

**SISTEM DETEKSI DAN PERHITUNGAN OTOMATIS BAKTERI
SALMONELLA DENGAN PENGOLAHAN CITRA
MENGUNAKAN METODE *OBJECT COUNTING***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh:

Lashot Ria Ingrid

NIM : 145150301111062



PROGRAM STUDI TEKNIK KOMPUTER
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

SISTEM DETEKSI DAN PERHITUNGAN OTOMATIS BAKTERI *SALMONELLA* DENGAN
PENGOLAHAN CITRA MENGGUNAKAN METODE *OBJECT COUNTING*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik

Disusun Oleh :
Lashot Ria Ingrid Melanika
145150301111062

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Juli 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Hurriyatul Fitriyah, S.T, M.Sc

NIP. 19851001 201504 2 003

Gembong Edhi Setyawan, S.T, M.T

NIK. 201208 761201 1 001

Mengetahui,

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T., M.T., Ph.D

NIP. 19710518 200312 1 001

PERNYATAAN ORISINILITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 26 Juli 2018



Lashot Ria Ingrid

NIM: 145150301111062

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yesus Kristus karena dengan berkat dan penyertaan-Nya, penulis dapat menyelesaikan skripsi dengan judul “Sistem Deteksi dan Perhitungan Otomatis Bakteri *Salmonella* dengan Pengolahan Citra Menggunakan Metode *Object Counting*” ini dapat terselesaikan. Laporan skripsi ini disusun dalam rangka memenuhi persyaratan untuk memperoleh gelar Sarjana Teknik di Fakultas Ilmu Komputer.

Penulis menyadari bahwa penyusunan laporan skripsi ini tidak lepas dari bantuan berbagai pihak baik secara langsung maupun tidak langsung. Dalam kesempatan ini penulis ingin menyampaikan ucapan terimakasih kepada:

1. Ayahanda penulis Linthon Damanik (Alm) yang semasa hidupnya selalu memberikan kepercayaan, nasihat, serta doa yang tidak pernah putus untuk penulis.
2. Ibunda penulis Helmina Sirait yang tidak pernah berhenti untuk memberikan dukungan secara moril dan materil, semangat dan juga doa.
3. Kedua saudara penulis Ruth Mora Adelia dan Jeremi Deardo Damanik yang selalu memberikan semangat.
4. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D. selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
5. Bapak Heru Nurwasito, Ir., M.Kom. selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Brawijaya Malang.
6. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D. selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang
7. Ibu Hurriyatul Fitriyah, S.T, M.Sc dan Bapak Gembong Edhi Setyawan, S.T, M.T selaku dosen pembimbing yang telah memberikan dukungan, saran, serta ilmu dalam membimbing penulis hingga pengerjaan skripsi ini selesai.
8. Segenap dosen dan staff Fakultas Ilmu Komputer Universitas Brawijaya atas ilmu pengetahuan serta nasihat dan bantuan administratif yang diberikan.
9. Sahabat penulis : Edo R Pratama, Daniel Sri Bintang, Dinda Agnes, Maria Sartika, Nella Sarah, Simson Hasian dan Victor Axelius yang selalu memberikan dukungan, keceriaan, serta semangat kepada penulis.
10. Terimakasih kepada Eva Agustyna, Tania Oka, Silvia Aprilla, Adelia Pinem, Dika Perdana, dan Joshua Kyrie untuk motivasi spiritual yang diberikan kepada penulis selama ini.
11. Keluarga besar PMK Daniel karena membantu penulis bertumbuh dalam hal-hal positif sehingga bisa menyelesaikan penelitian ini.
12. Teman-teman dekat penulis : Cindy Lilian, Shelsa F, Nafisa, Rizka Ayudya, Rarha, Bunga Boru, Olivia R Sitanggang, Enno Roscitra, Mulyahati Sutejo, Sara Yosephina, dan Ricky Guinta yang selalu membantu penulis dalam menyelesaikan penelitian ini.
13. The memories : Cynthia Veronica, Jessica Julia, Jessyca Tarida, Yohanes Tobing, Nadya Joan, Tesselonikha, Meylisan S, Karen Magdalena, Sharon Gabriella,

- Chelsea Gabrielle, Kristi Sri dan Kevin Benhard yang tak pernah putus-putusnya memberikan semangat dan dukungan kepada penulis.
14. Eksekutif Mahasiswa Teknik Komputer, terkhusus Departemen PSDM yang telah memberikan kesempatan untuk mengasah kemampuan diri selama menempuh studi di Teknik Komputer Universitas Brawijaya.
 15. Keluarga besar Teknik Komputer 2014 yang telah mendukung dan memberi arahan selama peneliti menempuh studi di Teknik Komputer Universitas Brawijaya dan selama penyelesaian skripsi.
 16. Seluruh pihak yang tidak dapat penulis sebutkan satu persatu dalam membantu dan memberikan arahan, penulis sangat mengucapkan terimakasih sebanyak-banyaknya.

Dengan segala keterbatasan pengetahuan yang dimiliki, penulis sadar bahwa laporan skripsi ini masih jauh dari kesempurnaan. Sehingga penulis berharap kritik dan saran yang membangun diberikan. Semoga skripsi ini dapat bermanfaat bagi penulis dan bagi semua pihak yang menggunakannya.

Malang, 26 Juli 2018

Penulis

inggridmelanica@gmail.com



ABSTRAK

Salmonella merupakan salah satu bakteri patogen yang dapat menyebabkan penyakit pada saluran pencernaan manusia. Ukuran yang mikroskopis membuat bakteri *Salmonella* tidak dapat dilihat tanpa bantuan mikroskop. Deteksi serta perhitungan koloni bakteri saat pengamatan dengan mikroskop masih dilakukan secara manual hingga saat ini. Maka dari itu dirancang sebuah sistem yang dapat mendeteksi dan melakukan perhitungan pada koloni bakteri *Salmonella* secara otomatis menggunakan metode segmentasi warna dan *object counting*. Sampel bakteri *Salmonella* hasil isolasi dari daging ayam dibuat pada kaca preparat dan dilakukan pewarnaan gram agar memudahkan saat diamati. Pengambilan gambar bakteri dilakukan menggunakan kamera pada mikroskop digital dengan resolusi 5 mega piksel. Proses pertama pada pengolahan citra yang dilakukan ialah *image enhancement*, kemudian konversi citra RGB ke citra HSV. Setelah itu dilakukan operasi morfologi *closing*, lalu koloni bakteri dihitung dengan metode *object counting*. Pemrosesan algoritma pengolahan citra dilakukan pada MATLAB, dan sistem akan ditampilkan pada sebuah antarmuka yang berguna untuk memudahkan penggunaan oleh *user*. Pengujian ketepatan *input* dan informasi gambar memperoleh hasil dengan status berhasil. Rerata waktu yang dibutuhkan saat eksekusi sistem ialah 4.59 detik, serta tingkat akurasi deteksi dan perhitungan jumlah koloni bakteri mempunyai tingkat akurasi sebesar 80.81%.

Kata kunci : Pengolahan Citra Digital, *Salmonella*, Mikroskop Digital, *Image Enhancement*, Segmentasi Warna, *Object counting*

ABSTRACT

Salmonella is one of the pathogenic bacteria that can cause disease in human digestive tract. The microscopic size makes the *Salmonella* bacteria invisible without the aid of a microscope. Detection and calculation of bacterial colonies during microscopic observations are still done manually until now. Therefore designed a system that can detect and perform calculations on *Salmonella* bacteria colonies automatically using the method of color segmentation and object counting. Samples of *Salmonella* bacteria isolated from chicken meat were made on glass preparations and gram staining was done to make it easier to observe. Bacteria shooting is using the camera on a digital microscope with a resolution of 5 mega pixels. The first process in image processing is image enhancement, then converting RGB image to HSV image. After that performed closing morphological operation, then bacterial colonies were calculated by object counting method. The processing of image processing algorithms is performed on MATLAB, and the system will be displayed on an interface to make it easier for user. The accuracy of input and image information to obtain results with successful status. The average time required during the execution of the system is 4.59 seconds, and the accuracy of detection and calculation of the number of bacterial colonies has an accuracy by 80.81%.

Keywords: Digital Image Processing, *Salmonella*, Digital Microscope, Image Enhancement, Color Segmentation, Object Counting

DAFTAR ISI

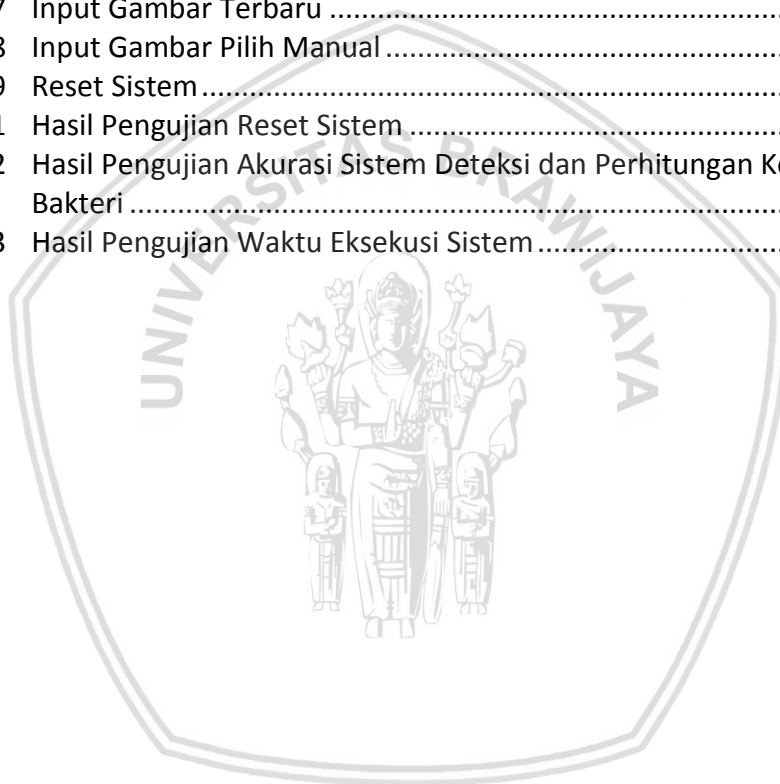
PENGESAHAN	i
PERNYATAAN ORISINILITAS.....	i
KATA PENGANTAR.....	iii
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Manfaat Penelitian	2
1.5 Batasan Penelitian	3
1.6 Sistematika Penulisan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka.....	5
2.2 Dasar Teori.....	6
2.2.1 Bakteri	6
2.2.2 <i>Salmonella</i>	7
2.2.3 Perhitungan Bakteri	7
2.2.4 Citra Digital.....	7
2.2.5 Pengolahan Citra Digital.....	7
2.2.6 Jenis Citra Digital	8
2.2.7 <i>Image Enhancement</i>	9
2.2.8 Segmentasi	10
2.2.9 Morfologi.....	11
2.2.10 <i>Object counting</i>	13
2.2.11 Mikroskop	14
2.2.12 MATLAB.....	14

2.2.13	GUI Pada MATLAB	15
2.2.14	Akurasi Sistem	15
BAB 3 METODOLOGI PENELITIAN		16
3.1	Alur Metode Penelitian	16
3.2	Studi Literatur	16
3.3	Analisis Kebutuhan	17
3.4	Perancangan Sistem	18
3.5	Implementasi Sistem	18
3.6	Pengujian dan Analisis	18
3.7	Pengambilan Kesimpulan	18
BAB 4 ANALISIS KEBUTUHAN		19
4.1	Kebutuhan Pengguna	19
4.1.1	Kebutuhan Antarmuka	20
4.2	Kebutuhan Sistem	20
4.2.1	Kebutuhan Perangkat Keras	21
4.2.2	Kebutuhan Perangkat Lunak	22
4.3	Kebutuhan Fungsional	23
4.4	Kebutuhan Non-Fungsional	23
4.4.1	Karakteristik Pengguna	23
4.4.2	Lingkungan Operasi	23
4.4.3	Asumsi dan Ketergantungan	24
4.4.4	Batasan Perancangan dan Implementasi	24
BAB 5 PERANCANGAN DAN IMPLEMENTASI		25
5.1	Data Sampel	26
5.1.1	Perancangan Data Sampel	26
5.1.2	Implementasi Data Sampel	26
5.2	Komunikasi Sistem	27
5.2.1	Perancangan Komunikasi Sistem	27
5.2.2	Implementasi Komunikasi Sistem	28
5.3	Deteksi dan Perhitungan Koloni Bakteri	28
5.3.1	Perancangan Deteksi dan Perhitungan Koloni Bakteri	28
5.3.2	Implementasi Deteksi dan Perhitungan Koloni Bakteri	33

5.4	Antarmuka Sistem	36
5.4.1	Perancangan Antarmuka Sistem	36
5.4.2	Implementasi Antarmuka Sistem	39
BAB 6 PENGUJIAN DAN ANALISIS		42
6.1	Pengujian Kebutuhan Fungsional	42
6.1.1	Pengujian <i>Input</i> Gambar	42
6.1.2	Pengujian Informasi Gambar	44
6.1.3	Pengujian <i>Reset</i> Program	45
6.2	Pengujian Akurasi Sistem	46
6.2.1	Tujuan Pengujian	46
6.2.2	Prosedur Pengujian	46
6.2.3	Hasil dan Analisis	47
6.3	Pengujian Waktu Eksekusi Sistem	48
6.3.1	Tujuan Pengujian	48
6.3.2	Prosedur Pengujian	48
6.3.3	Hasil dan Analisis	48
BAB 7 PENUTUP		50
7.1	Kesimpulan	50
7.2	Saran	50
DAFTAR PUSTAKA		51
LAMPIRAN		53

DAFTAR TABEL

Tabel 2.1	Warna dan Nilai Penyusun Warna	8
Tabel 4.1	Spesifikasi Laptop ASUS X450CP	21
Tabel 4.2	Spesifikasi Celestron LCD Digital Microscope II	22
Tabel 5.1	Implementasi high pass filter	33
Tabel 5.2	Konversi Citra RGB ke Citra HSV	34
Tabel 5.3	Konversi Citra Saturasi ke Citra Biner	34
Tabel 5.4	Operasi Morfologi Closing	35
Tabel 5.5	Deteksi dan Perhitungan Jumlah Koloni bakteri	36
Tabel 5.6	Penjelasan Sub Menu Pada Menu Utama	39
Tabel 5.7	Input Gambar Terbaru	40
Tabel 5.8	Input Gambar Pilih Manual	40
Tabel 5.9	Reset Sistem	41
Tabel 6.1	Hasil Pengujian Reset Sistem	46
Tabel 6.2	Hasil Pengujian Akurasi Sistem Deteksi dan Perhitungan Koloni Bakteri	47
Tabel 6.3	Hasil Pengujian Waktu Eksekusi Sistem	48



DAFTAR GAMBAR

Gambar 2.1	Diagram Blok Penelitian Sebelumnya	5
Gambar 2.2	Proses Morfologi Dilasi Pada Penelitian Sebelumnya.....	6
Gambar 2.3	Sistem Koordinat Citra Berukuran MxN	7
Gambar 2.4	Diagram Alir Tahap Pengolahan Citra Digital	8
Gambar 2.5	Citra RGB	8
Gambar 2.6	Citra Grayscale	9
Gambar 2.7	Citra Biner.....	9
Gambar 2.8	Representasi Sistem Warna HSV.....	11
Gambar 2.9	Citra Sebelum Dilasi dan Citra Sesudah Dilasi.....	12
Gambar 2.10	Citra Sebelum Erosi dan Citra Sesudah Erosi	13
Gambar 2.11	Citra Sebelum Operasi Closing dan Citra Sesudah Operasi Closing	13
Gambar 2.12	Mikroskop Digital	14
Gambar 2.13	Tampilan MATLAB.....	15
Gambar 3.1	Alur Metodologi Penelitian	16
Gambar 4.1	Diagram Analisis Kebutuhan	19
Gambar 4.2	Use Case Kebutuhan Pengguna	19
Gambar 4.3	Diagram Blok Sistem	20
Gambar 4.4	ASUS X450CP	21
Gambar 4.5	Celestron LCD Digital Microscope II	22
Gambar 5.1	Tahapan Perancangan dan Implementasi.....	25
Gambar 5.2	Laminar Air Flow	26
Gambar 5.3	Biakan Bakteri Salmonella dan Larutan Pewarnaan Gram	27
Gambar 5.4	Mikroskop Dengan Perbesaran Lensa Objektif 4x, 10x, dan 40x	27
Gambar 5.5	Diagram Blok Komunikasi Sistem.....	27
Gambar 5.6	Menyambungkan Mikroskop ke Laptop Dengan Kabel USB	28
Gambar 5.7	Alur Pendeteksian dan Perhitungan Jumlah Koloni bakteri	29
Gambar 5.8	Metode Zero Padding.....	30
Gambar 5.9	Letak Koordinat X,Y Pada Citra.....	31
Gambar 5.10	Hasil Nilai HSV dari koordinat yang sama	32
Gambar 5.11	Implementasi Image Enhancement	34
Gambar 5.12	Konversi Citra RGB ke Citra HSV.....	34
Gambar 5.13	Konversi citra saturasi ke citra biner.....	35
Gambar 5.14	Operasi Morfologi Closing.....	35
Gambar 5.15	Implementasi Deteksi dan Perhitungan Jumlah Koloni bakteri..	36
Gambar 5.16	Diagram Blok Perancangan Antarmuka Sistem	36
Gambar 5.17	Desain Antarmuka Sistem	37
Gambar 5.18	Membuat GUI Pada MATLAB	37
Gambar 5.19	Tampilan Awal Pembuatan GUI Pada MATLAB	38
Gambar 5.20	Tampilan Desain GUI	38
Gambar 5.21	Diagram Blok Perancangan Menu Utama Antarmuka Sistem	39
Gambar 5.22	Implementasi GUI	40

Gambar 6.1	Diagram Blok Pengujian Sistem	42
Gambar 6.2	Bakteri Dengan Perbesaran Lensa Objektif 4x.....	43
Gambar 6.3	Bakteri Dengan Perbesaran Lensa Objektif 10x.....	43
Gambar 6.4	Hasil Pengujian Input Gambar Terbaru.....	43
Gambar 6.5	Hasil Pengujian Input Gambar Secara Manual.....	44
Gambar 6.6	Informasi Input Gambar	45
Gambar 6.7	Hasil Perbandingan Waktu	45



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Ayam merupakan salah satu unggas yang paling banyak ditenak. Mudah dalam proses pengolahannya menjadikan daging ayam sebagai salah satu bahan pangan yang digemari banyak orang. Daging ayam memiliki kandungan gizi yang bernilai tinggi seperti protein, lemak, mineral yang dibutuhkan oleh tubuh. Harga daging ayam yang lebih ekonomis daripada daging sapi menjadi salah satu faktor yang membuat tinggi permintaan masyarakat akan daging ayam. Tapi seringkali terjadi kelalaian saat mengolah daging ayam, sehingga membuat daging ayam menjadi tercemar oleh bakteri.

Pada artikel yang ditulis oleh Sudarno (2016) mengatakan petugas dinas pertanian kota Bogor menemukan bahwa banyak daging ayam yang tercemar dengan bakteri karena dicuci menggunakan air sungai sebelum dijual. Bakteri yang ada meliputi *Salmonella* dan *E.Coli* dapat menyebabkan diare dan tifus. Hal ini membuktikan bahwa banyak sekali bakteri yang dapat mencemari daging ayam jika tidak dijaga kebersihannya. Ukuran *Salmonella* dan *E.Coli* sekitar $0.4 - 0.6 \times 1 - 3 \mu\text{m}$. Dengan ukuran mikroskopis tersebut sangat mustahil bakteri tersebut dapat dilihat tanpa bantuan mikroskop. Mikroskop mempunyai fungsi utama yaitu membesarkan benda yang dilihat sehingga membantu untuk mengamati benda yang renik (Lay, 1994).

Mikroskop mempunyai banyak jenis. Salah satunya ialah mikroskop digital. Mikroskop digital juga sering disebut mikroskop multimedia mempunyai fungsi yang sama seperti mikroskop pada umumnya. Perbedaannya ialah mikroskop digital dilengkapi dengan kamera digital dan gambar hasil pengamatan dapat diproyeksikan kedalam monitor komputer dan disimpan pada *file* komputer. Mikroskop sudah dilengkapi dengan kamera, memungkinkan untuk melakukan hal berbasis pengolahan citra.

Pengolahan citra merupakan istilah umum untuk menggambarkan berbagai teknik yang berguna untuk memanipulasi dan memodifikasi citra dengan berbagai cara (Efford, 2000). Pengolahan citra sudah menjadi bagian penting dalam kehidupan sehari-hari saat ini dan sudah diaplikasikan ke dalam berbagai aspek. Pada dunia kedokteran pengolahan citra mempunyai andil yang besar. Salah satu contoh pengaplikasian pengolahan citra pada bidang kedokteran ialah CT Scan. Selain itu pengolahan citra juga dapat digunakan dalam identifikasi atau pengenalan pola. Pengaplikasian pengolahan citra sedang dibutuhkan dalam bidang mikrobiologi untuk mendeteksi dan melakukan perhitungan pada objek mikroskopis seperti bakteri, mikroorganisme, dan darah secara otomatis.

Pada penelitian sebelumnya yang dilakukan Baby & Philip (2016) sudah berhasil untuk mendeteksi bakteri secara otomatis dengan menggunakan pengolahan citra. Begitu juga dengan penelitian yang dilakukan oleh Wibowo & Andrivani (2016) berhasil mendeteksi serta melakukan perhitungan jumlah koloni

bakteri pada setiap frame video menggunakan *counting morphology*. Kedua penelitian tersebut menggunakan bakteri *E.Coli* yang terdapat pada daging.

Dalam penelitian ini menggunakan bakteri *Salmonella* yang terdapat pada daging ayam, karena belum ada penelitian yang menggunakan bakteri tersebut sebagai objek. Sehingga akan diterapkan sebuah sistem deteksi dan perhitungan otomatis jumlah koloni bakteri *Salmonella* pada sampel daging ayam menggunakan segmentasi warna dan *object counting*. Penggunaan *object counting* merupakan salah satu metode yang digunakan untuk melakukan perhitungan jumlah koloni bakteri secara otomatis. Penggunaan *object counting* juga bertujuan untuk mempercepat perhitungan koloni bakteri yang masih dilakukan secara manual oleh laboran. Pada penelitian ini juga sistem menggunakan GUI (*Graphical user interface*) sehingga dapat memudahkan laboran dalam mendeteksi dan melakukan perhitungan saat melakukan pengamatan pada bakteri *Salmonella* dengan mikroskop digital.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dari itu dapat dirumuskan masalah penelitian sebagai berikut:

1. Bagaimana sistem dapat mendeteksi dan menghitung jumlah koloni bakteri *Salmonella* secara otomatis menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*?
2. Bagaimana implementasi *user interface* untuk deteksi dan perhitungan otomatis bakteri *Salmonella* dengan menggunakan MATLAB?
3. Bagaimana tingkat keberhasilan sistem dalam mendeteksi dan melakukan perhitungan otomatis bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*?
4. Berapa lama waktu yang dibutuhkan saat mengeksekusi sistem?

1.3 Tujuan Penelitian

Mengacu pada rumusan masalah yang sudah dijabarkan, maka penelitian ini dibuat dengan tujuan sebagai berikut:

1. Untuk mengetahui bagaimana sistem dapat mendeteksi dan menghitung jumlah koloni bakteri *Salmonella* secara otomatis menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.
2. Untuk mengetahui implementasi dari *user interface* untuk sistem deteksi dan perhitungan otomatis bakteri *Salmonella* dengan menggunakan MATLAB.
3. Mampu menguji tingkat keberhasilan deteksi dan perhitungan otomatis bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.
4. Untuk mengetahui waktu yang dibutuhkan saat mengeksekusi sistem dan *reset* sistem.

1.4 Manfaat Penelitian

Dalam penelitian ini diharapkan dapat memberi beberapa manfaat seperti:

1. Memberikan pengetahuan lebih luas terhadap pembaca terkait penelitian sistem deteksi dan perhitungan otomatis bakteri *Salmonella* dengan pengolahan citra menggunakan metode segmentasi warna dan *object counting* menggunakan MATLAB.
2. Memungkinkan pendeteksian dan perhitungan koloni bakteri *Salmonella* secara otomatis menggunakan mikroskop digital.
3. Sebagai bentuk penelitian terkait sistem deteksi dan perhitungan otomatis koloni bakteri *Salmonella* yang nantinya bisa dikembangkan lebih lanjut oleh peneliti lain baik dengan objek ataupun metode yang sama ataupun berbeda.

1.5 Batasan Penelitian

Berikut ini merupakan batasan yang ada pada penelitian ini agar permasalahan yang dirumuskan dapat lebih fokus:

1. Sampel daging ayam yang digunakan ialah daging ayam potong pada pasar tradisional
2. Bakteri yang dideteksi dan dihitung ialah Bakteri *Salmonella*.
3. Bakteri *Salmonella* sudah diwarnai dulu dengan teknik pewarnaan gram.
4. Bakteri yang bergandengan dianggap sebagai satu koloni bakteri.
5. Ukuran dari bakteri *Salmonella* diabaikan.
6. Menggunakan bakteri dengan usia 20 – 24 jam.
7. Penelitian dilakukan hanya menghitung jumlah koloni bakteri yang terdapat pada gambar tanpa ada proses lanjutan.

1.6 Sistematika Penulisan

Sistematika pembahasan dari penulisan ini menggambarkan bagaimana urutan dan uraian dari laporan tugas akhir secara garis besar sebagai berikut:

BAB 1 : PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan, serta sistematika penelitian laporan skripsi.

BAB 2 : LANDASAN KEPUSTAKAAN

Bab ini berisi tentang teori-teori yang mendukung penelitian sistem deteksi dan perhitungan bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.

BAB 3 : METODE PENELITIAN

Bab ini berisi metode penelitian yang akan digunakan dalam implementasi dan pengujian sistem deteksi dan perhitungan bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.

BAB 4 : ANALISIS KEBUTUHAN

Bab ini berisi penjelasan tentang kebutuhan apa saja yang dibutuhkan oleh sistem, baik kebutuhan fungsional maupun kebutuhan non-fungsional.

BAB 5 : PERANCANGAN DAN IMPLEMENTASI

Bab ini berisi tentang tahap-tahap perancangan dan implementasi sistem deteksi dan perhitungan bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.

BAB 6 : PENGUJIAN DAN ANALISIS

Bab ini berisi tentang pengujian serta analisis terhadap sistem deteksi dan perhitungan bakteri *Salmonella* menggunakan mikroskop digital dengan metode segmentasi warna dan *object counting*.

BAB 7 : PENUTUP

Bab ini berisi kesimpulan dan saran yang diperoleh dari hasil pengujian dan analisis sistem agar dapat dikembangkan dikemudian hari, serta berguna untuk mengetahui apakah terdapat kesalahan pada implementasi dari hasil penelitian.



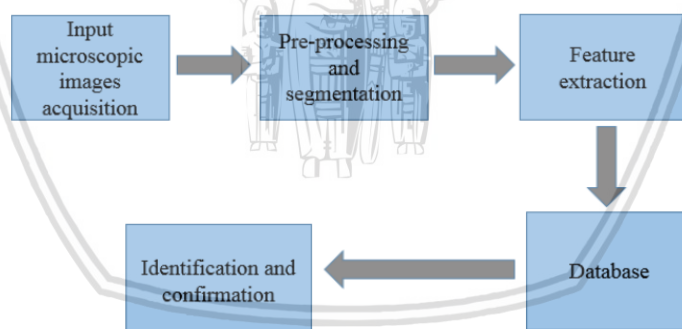
BAB 2 LANDASAN KEPUSTAKAAN

Pada bab ini berisi penjelasan kajian kepustakaan yang digunakan untuk penelitian sistem deteksi dan perhitungan otomatis koloni bakteri *Salmonella*. Kajian yang terdapat pada bab ini juga membahas penelitian sebelumnya yang memiliki keterkaitan dengan penelitian yang dilakukan oleh penulis.

2.1 Kajian Pustaka

Kajian pustaka berisi tentang referensi yang digunakan pada penelitian saat ini dan mempunyai hubungan baik dari metode ataupun objek yang digunakan dalam penelitian sebelumnya. Dengan adanya referensi dari penelitian sebelumnya maka akan memudahkan untuk membenahi kekurangan yang ada dan bisa membantu penelitian yang akan dibuat kedepannya.

Pada penelitian Baby & Philip (2016) menjelaskan tentang penggunaan *low-cost embedded* dalam pengolahan citra mikroskopis. Penelitian tersebut menggunakan Raspberry Pi sebagai mikrokomputernya. Bakteri yang digunakan ialah bakteri *E.Coli* yang diberi teknik pewarnaan gram. Metode yang digunakan ialah segmentasi warna. Penelitian ini menggunakan mikroskop *electron* yang dilengkapi kamera untuk akuisisi citra dan menggunakan bahasa pemrograman python. Pada penelitian ini tidak digunakan teknik *image enhancement* untuk memperbaiki kualitas citra. Begitu juga dengan bakteri yang terdeteksi tidak dihitung secara otomatis oleh sistem, sehingga *user* harus menghitung secara manual. Keseluruhan sistem yang dilakukan dapat dilihat pada Gambar 0.1

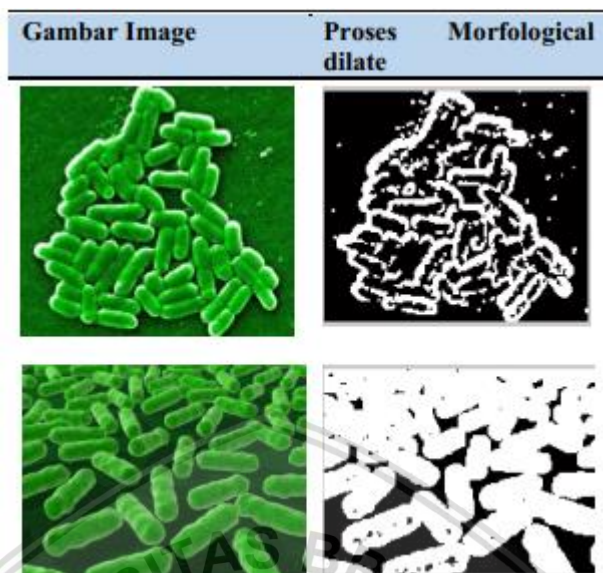


Gambar 0.1 Diagram Blok Penelitian Sebelumnya

Sumber : (Baby & Philip, 2016)

Penelitian lainnya yang dilakukan oleh Wibowo & Andrivani (2016) ialah perhitungan koloni bakteri *E.coli* dengan menggunakan metode *thresholding* dan *counting morphology*. Pada penelitian ini dijelaskan bahwa metode *counting morphology* dapat digunakan untuk menghitung jumlah koloni bakteri pada setiap frame video. Penelitian ini menggunakan metode morfologi dilasi. Pada penelitian ini tidak ada fungsi yang digunakan untuk membaca data secara langsung sehingga mengurangi fleksibilitas sistem, serta tidak membandingkan hasil perhitungan sistem dengan hasil perhitungan dari pakar. Selain itu Pada penelitian ini juga

belum menggunakan GUI untuk mempermudah penggunaan sistem. Pada Gambar 0.2 dapat dilihat salah satu metode yang digunakan pada penelitian ini.



Gambar 0.2 Proses Morfologi Dilasi Pada Penelitian Sebelumnya

Sumber : (Wibowo & Andrivani, 2016)

Maka dari itu, penelitian kali ini juga menerapkan hal yang sama dalam pendeteksian dan perhitungan koloni bakteri secara otomatis berbasis pengolahan citra. Bakteri yang diamati pada penelitian ini adalah bakteri *Salmonella* pada daging ayam. Namun, pada sistem ini menggunakan metode *image enhancement*, segmentasi warna dalam memisahkan koloni bakteri dengan objek yang lain dan metode *object counting* untuk menghitung jumlah koloni bakteri yang ada dalam gambar. Kemudian hasil akan ditampilkan dalam bentuk GUI untuk mempermudah penggunaan sistem oleh *user*.

2.2 Dasar Teori

2.2.1 Bakteri

Bakteri umumnya berbentuk sel tunggal atau uniseluler, tidak mempunyai klorofil berkembangbiak dengan pembelahan sel atau biner. Karena tidak mempunyai klorofil, bakteri hidup sebagai jasad yang saprofitik ataupun sebagai jasad yang parasitik. Tempat hidupnya tersebar di mana-mana, yaitu di udara, di dalam tanah, didalam air, pada bahan-bahan, pada tanaman ataupun pada tubuh manusia atau hewan (Putri, Sukini, & Yodong, 2018).

Bakteri dibedakan menjadi tiga melalui karakteristik dinding selnya, yaitu bakteri gram positif, bakteri gram negatif, dan bakteri tidak berdinding sel. bakteri gram positif adalah bakteri yang dinding selnya menyerap warna violet serta memiliki lapisan peptidoglikan. Sedangkan bakteri gram negatif adalah bakteri yang dinding selnya menyerap warna merah serta lapisan peptidoglikan yang dimiliki tipis.

2.2.2 *Salmonella*

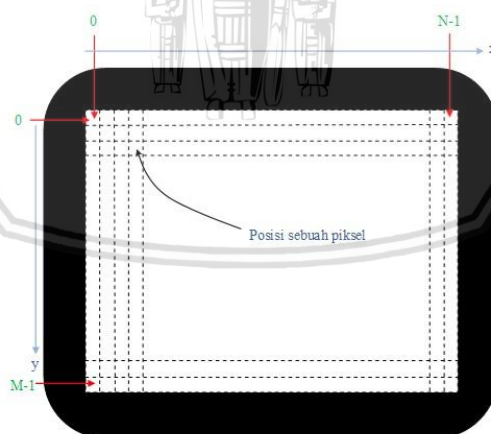
Salmonella merupakan genus bakteri enterokoloni bakteri yang masuk kedalam jenis bakteri gram negatif dan berbentuk tongkat, serta merupakan bakteri patogen. *Salmonella* dapat menyebabkan tifoid, paratifoid, dan juga penyakit *foodborne* (Poeloengan, Komala, & Noor). *Salmonella* menyebabkan penyakit pada organ bagian pencernaan. Penyakit yang disebabkan oleh *Salmonella* antara lain adalah diare, keram perut, dan demam dalam jangka waktu 8-72 jam setelah memakan makanan yang terkontaminasi *Salmonella*.

2.2.3 Perhitungan Bakteri

Perhitungan bakteri dibedakan menjadi dua, yaitu perhitungan secara langsung dan tidak langsung. Perhitungan bakteri secara langsung mempunyai beberapa metode, yaitu seperti metode hitungan cawan (total plate count) yang menghitung jumlah koloni bakteri pada cawan petri dengan menggunakan teknik pengenceran. Pada teknik tersebut bakteri yang berdekatan dapat dihitung sebagai satu koloni bakteri (Fardiaz, 1993).

2.2.4 Citra Digital

Citra digital merupakan gambar yang terdiri dari dua dimensi yang bisa ditampilkan pada layar komputer sebagai sebuah himpunan (kumpulan) digital atau yang sering disebut dengan *pixel*. Setiap *pixel* memiliki titik koordinat posisi. Citra digital dibentuk melalui pendekatan kuantisasi. Kuantisasi merupakan prosedur untuk mengubah isyarat yang bersifat kontinu kedalam bentuk diskret (Kadir & Susanto, 2013). Sistem koordinat yang dipakai untuk menyatakan citra digital dapat dilihat pada Gambar 0.3.



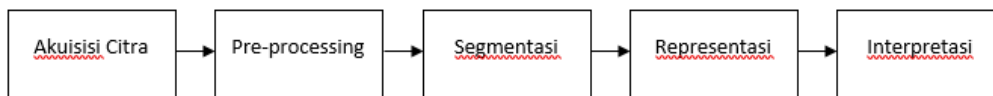
Gambar 0.3 Sistem Koordinat Citra Berukuran $M \times N$

Sumber : (Kadir & Susanto, 2013)

2.2.5 Pengolahan Citra Digital

Pengolahan citra digital ialah penggunaan algoritma komputer untuk melakukan pemrosesan pada gambar digital. Dengan kata lain pengolahan citra digital juga merupakan teknik pemrosesan gambar sampai menghasilkan gambar

yang sesuai dengan keinginan (Pamungkas, Adi, & Anam, 2014). Secara garis besar langkah-langkah dalam melakukan pengolahan citra digital dapat dilihat pada Gambar 0.4.



Gambar 0.4 Diagram Alir Tahap Pengolahan Citra Digital

2.2.6 Jenis Citra Digital

Ada beberapa jenis citra yang sering dipakai dalam proses pengolahan citra digital. Jenis citra tersebut memiliki karakteristik yang berbeda satu sama lain. Citra RGB, citra *grayscale* (keabuan), citra biner merupakan beberapa jenis citra tersebut.

2.2.6.1 Citra RGB

Citra RGB atau yang juga sering disebut sebagai gambar *truecolor* merupakan jenis citra yang disimpan dalam *array* data dan menyajikan warna dalam bentuk komponen R (*red*), G (*green*), B (*blue*) pada masing-masing *pixel*. Warna pada tiap *pixel* ditentukan oleh kombinasi dari intensitas nilai R, G, dan B dan masing-masing komponen warna tersebut memiliki nilai 8 bit (Mathworks, 2018). Pada Tabel 0.1 dan Gambar 0.5 menunjukkan contoh warna dan nilai R, G, dan B.

Tabel 0.1 Warna dan Nilai Penyusun Warna

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255



Gambar 0.5 Citra RBG

Sumber : (Mathworks, 2018)

2.2.6.2 Citra Grayscale

Citra keabuan sering juga disebut dengan citra *grayscale* menangani gradasi warna hitam dan putih yang menghasilkan warna dengan efek abu-abu (Kadir & Susanto, 2013). Intensitas gambar berkisar antara 0 sampai 255 dimana nilai 0

menyatakan hitam dan nilai 255 menyatakan putih. Untuk mengubah citra RGB ke citra *grayscale* menggunakan nilai rerata dari tiap komposisi warna R, G, dan B. Pada Gambar 0.6 merupakan citra *grayscale*.

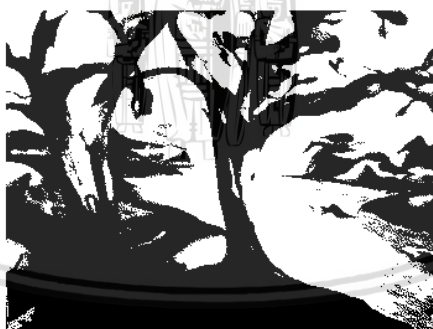


Gambar 0.6 Citra Grayscale

Sumber : (Mathworks, 2018)

2.2.6.3 Citra Biner

Citra biner merupakan citra dengan dua kemungkinan intensitas warna disetiap pixelnya yaitu 0 dan 1 (Kadir & Susanto, 2013). Nilai 0 menyatakan warna hitam dan nilai 1 menyatakan warna putih. Citra biner sering digunakan dalam pemrosesan citra untuk kepentingan memperoleh tepi bentuk suatu objek. Jika citra RGB bernilai 8 bit disetiap pixelnya, maka citra biner hanya bernilai 1 bit disetiap *pixel*nya. Pada Gambar 0.7 merupakan hasil dari citra RGB yang dikonversi ke citra biner.



Gambar 0.7 Citra Biner

Sumber : (Mathworks, 2018)

Konversi gambar asli ke dalam gambar biner mempunyai beberapa variabel, dimana variabel $b(i)$ merupakan nilai biner yang ingin dicari. Konversi tersebut dapat dilakukan dengan persamaan sebagai berikut :

$$b(i) = \begin{cases} 0, & i < 128 \\ 255, & i \geq 128 \end{cases} \quad (2.1)$$

2.2.7 Image Enhancement

Image enhancement atau perbaikan kualitas citra merupakan bagian dari tahapan *pre-processing* yang umumnya dilakukan sebelum tahapan segmentasi.

Perbaikan kualitas citra dilakukan pada citra yang mempunyai kualitas buruk seperti adanya *noise*, citra terlalu gelap ataupun terang, citra terlalu kabur, dan lain sebagainya. Perbaikan kualitas citra digunakan untuk menonjolkan ciri citra dan mempermudah saat mengekstraksi dan melakukan analisis citra pada proses segmentasi. Ada beberapa teknik *image enhancement* yang sering digunakan yaitu operasi spasial, contohnya mengubah kontras, *window slicing*, dan pemotongan *noise*. Operasi spasial, contohnya *smoothing noise*, *low-pass*, *band-pass*, dan *high-pass* filter. Serta operasi transformasi, contohnya *linier*, *root*, dan *homomorphic* filter.

Operasi spasial merupakan teknik untuk memodifikasi *pixel* berdasarkan *pixel* yang dipilih dan *pixel* tetangganya. Operasi spasial membutuhkan kernel berupa matrik bujur sangkar seperti 3x3 dan 5x5 (Kadir & Susanto, 2013). Operasi spasial didefinisikan dengan persamaan berikut dengan $x(i, j)$ sebagai *image input* dan $h(u, v)$ sebagai kernel. Proses perkalian antara kernel dan *input image* sering disebut sebagai konvolusi.

$$y(i, j) = \sum_v \sum_u h(u, v) \cdot x(i - u, j - v) \quad (2.2)$$

High Pass Filter merupakan salah satu contoh operasi spasial. *High Pass Filter* digunakan untuk melakukan proses deteksi tepi. Ciri dari *high pass filter* ialah nilainya terdiri dari nilai nol, positif, negatif, dan jumlah dari semua nilainya sama dengan nol seperti yang terlihat pada persamaan berikut.

$$\sum_{uv} H(u, v) = 0 \quad (2.3)$$

Salah satu kernel pada *high pass filter* ialah terdapat kernel tiga yang digunakan pada penelitian ini dapat dilihat pada persamaan berikut:

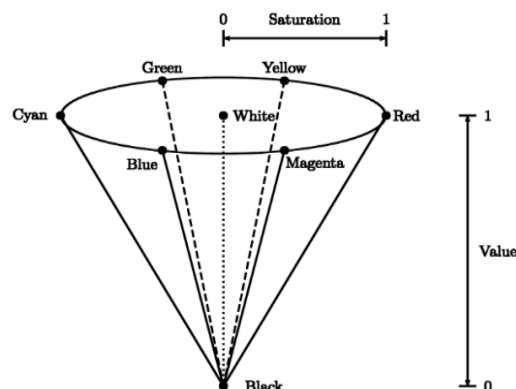
$$K3 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix}, \text{dimana } \sum = 1 \quad (2.4)$$

2.2.8 Segmentasi

Segmentasi merupakan proses yang ditujukan untuk mendapatkan objek yang terkandung dalam citra dengan kata lain membagi citra ke dalam beberapa daerah yang memiliki kemiripan atribut (Kadir & Susanto, 2013). Berdasarkan teknik yang digunakan, menurut Rangayyan (2015) segmentasi dibagi kedalam empat kategori yaitu teknik pengambangan (*threshold*), metode berbasis batas, metode berbasis area, dan metode hibrid yang mengkombinasikan kriteria batas dan area.

2.2.8.1 Segmentasi Warna Berdasar Komponen HSV

HSV (*hue, saturation, value*) merupakan salah satu sistem warna yang sering digunakan pada saat akan memilih warna (Sianipar, Mangiri, & Wiryajati, 2013). Gambar 0.8 merupakan representasi ruang warna sistem HSV.



Gambar 0.8 Representasi Sistem Warna HSV

Sumber : (Sianipar, Mangiri, & Wiryajati, 2013)

Jika warna R, G, dan B memiliki nilai yang sama maka warna akan menjadi keabuan dan membuat nilai *saturation* menjadi 0, begitu juga sebaliknya. Sedangkan *hue* didefinisikan sebagai nilai sudut 0° sampai 360°. Perhitungan konversi dari citra RGB ke ruang warna dapat dilihat seperti persamaan berikut.

$$R' = \frac{R}{255} \quad (2.5)$$

$$G' = \frac{G}{255} \quad (2.6)$$

$$B' = \frac{B}{255} \quad (2.7)$$

Kemudian mencari nilai *hue*, *saturation*, dan *value* menggunakan persamaan berikut:

$$V = \max\{R, G, B\} \quad (2.8)$$

$$\delta = V - \min\{R, G, B\} \quad (2.9)$$

$$S = \frac{\delta}{V} \quad (2.10)$$

$$H = \begin{cases} 60^\circ \times \left(\frac{G' - B'}{\delta} \bmod 6 \right), & Vmax = R' \\ 60^\circ \times \left(\frac{B' - R'}{\delta} + 2 \right), & Vmax = G' \\ 60^\circ \times \left(\frac{R' - G'}{\delta} + 4 \right), & Vmax = B' \end{cases} \quad (2.11)$$

Variabel V digunakan untuk mencari nilai *value* berdasarkan perbandingan nilai maksimum antara R, G, dan B. Variabel δ digunakan untuk mencari selisih antara nilai maksimum dan minimum antara R, G, dan B. Sedangkan variabel S ialah untuk mencari nilai saturasi, dan variabel H ialah untuk mencari nilai *hue*.

2.2.9 Morfologi

Operasi morfologi adalah operasi umum dalam pengolahan citra yang secara spesifik digunakan untuk menganalisis bentuk dalam citra. Operasi morfologi juga mampu untuk mengubah struktur bentuk objek yang terkandung didalam citra (Kadir & Susanto, 2013). Dua operasi yang mendasari operasi morfologi ialah

dilasi dan erosi, lalu dikembangkan lagi menjadi operasi morfologi *closing* dan *opening* yang dibentuk melalui operasi dasar tersebut.

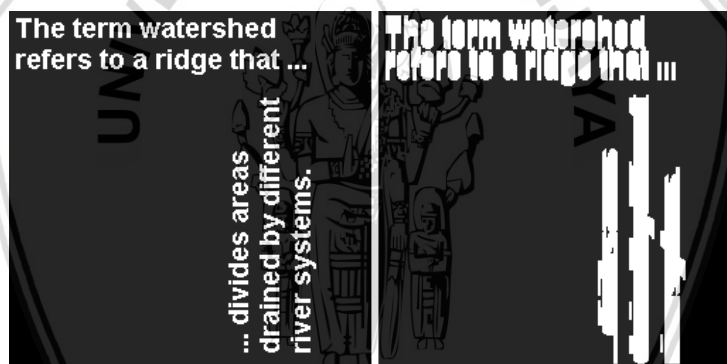
Operasi morfologi melibatkan dua larik *pixel*. Larik pertama berupa citra yang akan di morfologi dan larik kedua dinamakan kernel atau *structuring element* (Shih, 2009). *Structuring element* digunakan sebagai parameter dalam mengurangi dan menambahkan jumlah *pixel*. *Structure element* adalah sebuah gambar biner dengan nilai 1 sebagai penunjuk bentuk yang akan dioperasikan. Bentuk *structuring element* yang biasa digunakan ialah *rectangle*, *square*, *disk*, *diamond*, dan *linear*.

2.2.9.1 Dilasi

Dilasi merupakan operasi yang digunakan untuk memperbesar ukuran segmen objek dengan cara menambahkan ruang ditepi objek. Hasil dari operasi dilasi ialah gambar menjadi lebih tebal daripada gambar sebelumnya. Rumus dilasi dapat dilihat pada persamaan berikut:

$$A \oplus B = \bigcup_{b \in B} A_b \quad (2.12)$$

Variabel A ialah citra *input* sedangkan B adalah *structuring element*. Contoh proses dilasi ialah pada Gambar 0.9.



Gambar 0.9 Citra Sebelum Dilasi dan Citra Sesudah Dilasi

Sumber : (Mathworks, 2018)

2.2.9.2 Erosi

Operasi erosi mempunyai efek memperkecil struktur citra. Operasi erosi menggunakan intersection antara gambar A dengan *structure element* yang digunakan. Berikut ini persamaan dari operasi erosi.

$$A \ominus B = \bigcap_{b \in B} A_{-b} \quad (2.13)$$

Dimana *structure element* B memenuhi setiap bagian dari gambar asal atau A maka nilai *pixel* di A akan dipertahankan. Pada Gambar 0.10 merupakan operasi hasil erosi.



Gambar 0.10 Citra Sebelum Erosi dan Citra Sesudah Erosi

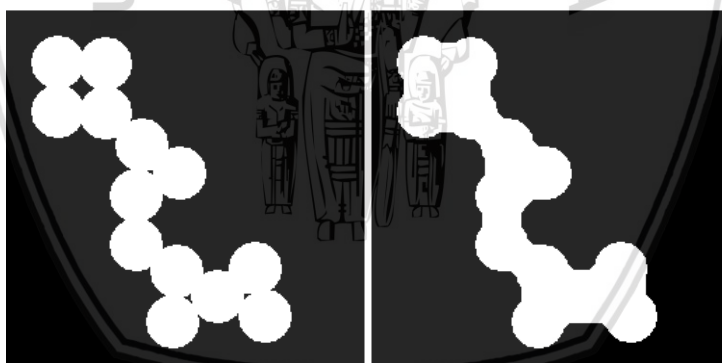
Sumber : (Mathworks, 2018)

2.2.9.3 Closing

Operasi *closing* adalah kebalikan dari operasi *opening*. Citra pada operasi *closing* akan di dilasi terlebih dahulu lalu hasilnya dilanjutkan dengan erosi. Operasi *closing* berguna untuk menghilangkan lubang-lubang kecil dan juga menghaluskan kontur citra (Kadir & Susanto, 2013). Persamaan dari operasi *closing* ialah sebagai berikut.

$$A \circ B = (A \oplus B) \otimes B \quad (2.14)$$

Pada dasarnya operasi *closing* dilakukan untuk menggabungkan objek yang berdekatan. Berikut pada Gambar 0.11 contoh operasi *closing*.



Gambar 0.11 Citra Sebelum Operasi Closing dan Citra Sesudah Operasi Closing

Sumber : (Mathworks, 2018)

2.2.10 Object counting

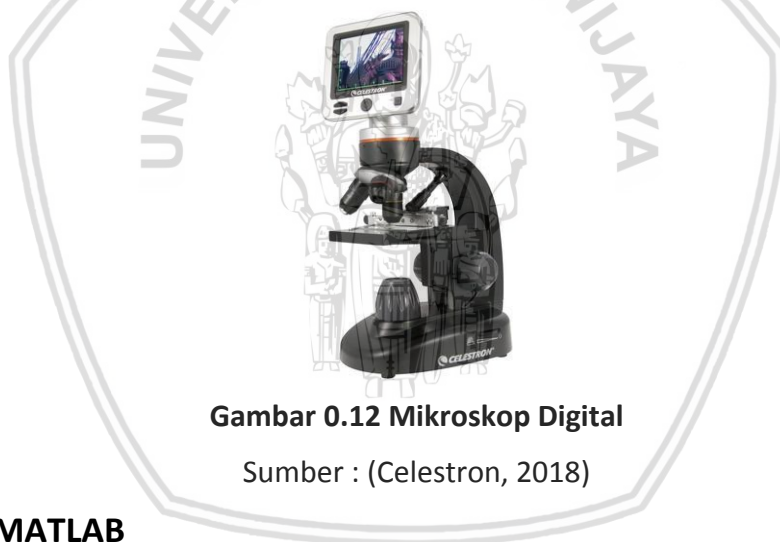
Object counting merupakan sebuah metode yang digunakan dan dikembangkan untuk menghitung objek yang terdapat pada citra dengan menggunakan *regionprop* dan *centroid*, dan menentukan objek yang akan diberikan *boundingbox*. Pada metode ini digunakan citra hasil morfologi kemudian sistem akan mengkalkulasi berapa jumlah koloni bakteri yang terdapat pada *input* gambar. Beberapa variabel yang digunakan ialah *regionprops*, *centroid*, *boundingbox*.

2.2.11 Mikroskop

Mikroskop merupakan alat utama yang selalu digunakan pada laboratorium mikrobiologi. Mikroskop memiliki fungsi untuk memperbesar benda yang dilihat sehingga dapat membantu untuk mengamati dan memperhatikan benda yang renik (Lay, 1994). Mikroskop ditemukan oleh Anthony Van Leewenhoek. Bagian-bagian dari mikroskop dibedakan menjadi dua yaitu bagian optik dan non optik. Mikroskop mempunyai banyak jenis yang dapat dibedakan dari jumlah lensa, sumber cahaya, dan juga berdasarkan nama serta kegunaannya. Salah satu contoh mikroskop yang banyak digunakan pada masa sekarang ini alah mikroskop digital.

2.2.11.1 Mikroskop Digital

Mikroskop digital merupakan perpaduan antara mikroskop optik dan juga kamera digital. Penggunaan mikroskop digital sangat mudah dan tidak harus menggunakan mikroskop didekat sumber cahaya, karena sudah dilengkapi dengan lampu. Mikroskop digital juga sering disebut sebagai mikroskop multimedia, dikarenakan selama melakukan proses penelitian mikroskop ini mampu untuk merekam dan mengambil gambar objek yang diteliti. Gambar 0.12 adalah salah satu mikroskop digital saat ini.

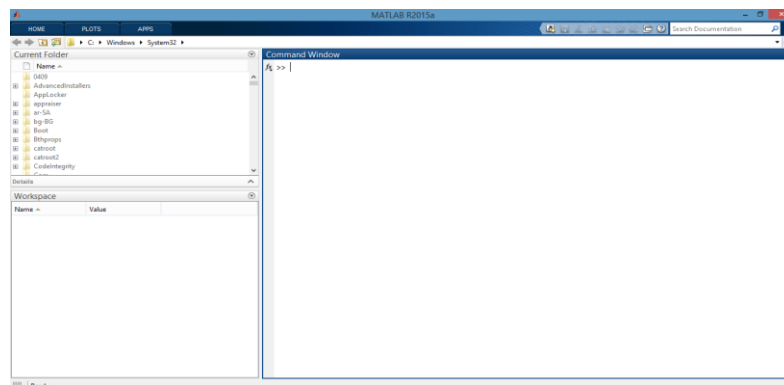


Gambar 0.12 Mikroskop Digital

Sumber : (Celestron, 2018)

2.2.12 MATLAB

Matrix Laboratory atau yang biasa disebut MATLAB merupakan bahasa pemrograman dengan kemampuan tinggi dalam bidang memproses operasi matematis dan visualisasi (Sugiarto, 2006). MATLAB menggunakan matriks sebagai dasar komputasinya. MATLAB membagi matriks menjadi dua yaitu matriks khusus dan matriks yang didefinisikan oleh *user*. Contoh matriks khusus ialah matriks diagonal, matriks pascal, matriks random, matriks identitas, matriks satu, dan matriks *magic square*. MATLAB juga dapat digunakan untuk pemrosesan sistem kontrol, pengolahan citra, serta pengenalan pola. Tampilan aplikasi MATLAB dapat dilihat pada Gambar 0.13 berikut.



Gambar 0.13 Tampilan MATLAB

Aplikasi MATLAB juga mempunyai fungsi untuk menampilkan grafik serta kurva dan menggunakan antarmuka C++, Java, NET, Python, SQL, serta Microsoft Excel.

2.2.13 GUI Pada MATLAB

MATLAB juga menyediakan sebuah pemrograman berorientasi visual atau lebih dikenal dengan *graphical user interface* (GUI) yang dapat dibangun dengan objek-objek grafis seperti tombol, *slider*, kotak teks, maupun menu (Sugiarto, 2006).

2.2.14 Akurasi Sistem

Persentase keberhasilan sistem digunakan untuk mengetahui apakah sistem sudah sesuai dengan yang diinginkan atau belum. Persentase keberhasilan sistem juga bertujuan untuk mengevaluasi algoritma serta metode yang digunakan. Rumus yang digunakan dalam menghitung persentase keberhasilan sistem terdapat pada persamaan berikut:

$$\text{Persentase keberhasilan} = \frac{A - |A - B|}{A} \times 100\% \quad (2.15)$$

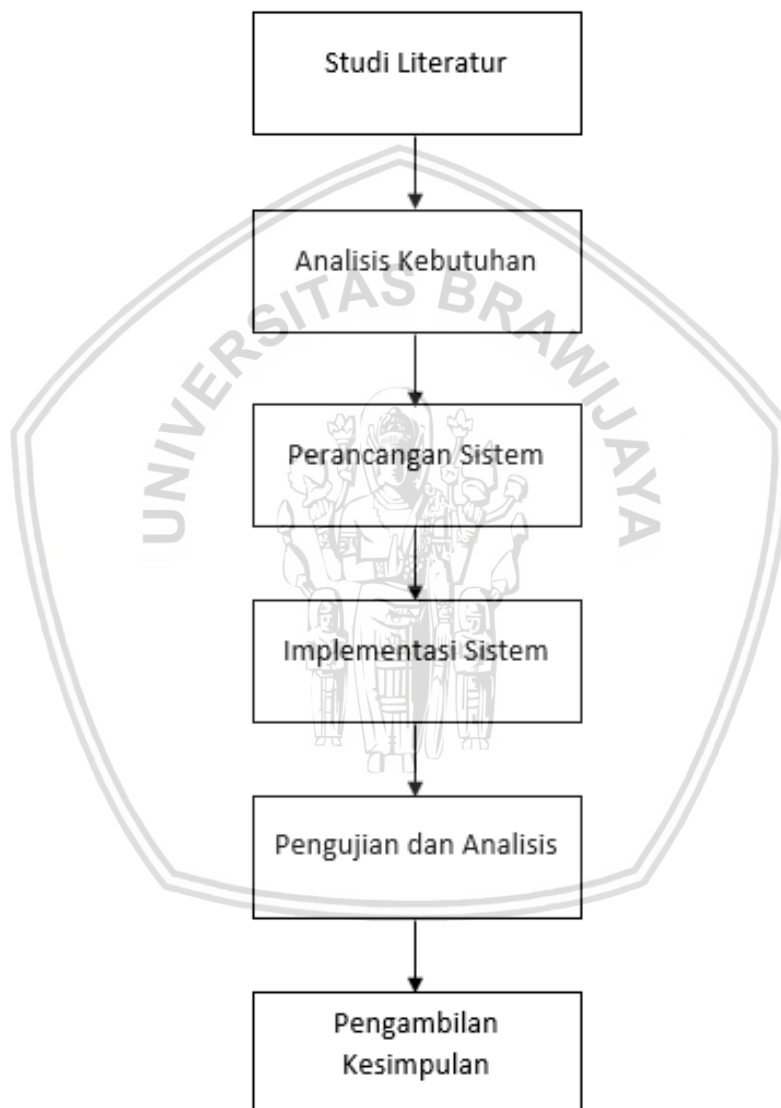
A adalah nilai yang diharapkan, dan B adalah nilai dari hasil pengujian. Dalam menghitung persentase keberhasilan dibutuhkan banyak percobaan dalam menentukan perbedaan antara nilai riil dengan nilai pembacaan. Maka dari itu dibutuhkan perhitungan rata-rata terhadap seluruh percobaan yang dilakukan. Berikut ini persamaan yang digunakan:

$$\text{Rata - rata} = \frac{\text{Jumlah Nilai}}{\text{Banyak Data}} \quad (2.16)$$

BAB 3 METODOLOGI PENELITIAN

3.1 Alur Metode Penelitian

Penelitian yang dilakukan adalah penelitian implementatif. Penelitian dimulai dari studi literatur, analisis kebutuhan, perancangan, implementasi, pengujian, dan analisis sistem, serta pengambilan keputusan. Pada Gambar 0.1 ini merupakan *flowchart* yang menggambarkan alur metodologi penelitian ini secara umum.



Gambar 0.1 Alur Metodologi Penelitian

3.2 Studi Literatur

Studi literatur merupakan proses mempelajari literatur yang bersumber dari jurnal, artikel, buku, maupun penelitian sebelumnya digunakan untuk memperkuat dasar penulisan penelitian. Teori-teori yang dijadikan sebagai bahan pendukung penelitian ini ialah :

1. Bakteri
Menjelaskan pengertian bakteri dan perbedaan bakteri
2. *Salmonella*
Menjelaskan tentang struktur, ukuran serta penyebab yang ditimbulkan oleh *Salmonella*.
3. Perhitungan Bakteri
Menjelaskan tentang cara perhitungan bakteri secara manual
4. Citra Digital
Menjelaskan apa itu citra digital dan komponen apa saja yang membentuk citra digital
5. Pengolahan Citra Digital
Menjelaskan apa itu pengolahan citra digital dan jenis dari pengolahan citra digital, serta tahap-tahap apa saja yang dilakukan dalam pengolahan citra digital.
6. Jenis Citra
Menjelaskan perbedaan citra berdasarkan nilai *pixel* yaitu citra RGB, citra *grayscale*, dan citra biner.
7. *Image Enhancement*
Menjelaskan tentang perbaikan kualitas citra pada bagian *pre-processing* dengan menggunakan teknik *high pass filter*.
8. Segmentasi
Menjelaskan mengenai pengertian segmentasi dan contoh-contoh segmentasi seperti segmentasi komponen HSV.
9. Morfologi
Menjelaskan tentang proses morfologi serta metode morfologi yang ada seperti dilasi, erosi, *closing*, dan *opening*.
10. *Object counting*
Menjelaskan tentang metode *object counting*.
11. Mikroskop
Menjelaskan apa itu mikroskop, bagian-bagian mikroskop, serta beberapa jenis mikroskop yang sering digunakan.
12. MATLAB
Menjelaskan apa itu MATLAB mulai dari kegunaan hingga kekurangan dan kelebihan.
13. GUI
Menjelaskan tentang GUI atau *Graphic User Interface*.
14. Akurasi Sistem
Menjelaskan tentang perhitungan akurasi sistem.

3.3 Analisis Kebutuhan

Tahap ini digunakan untuk menganalisis kebutuhan apa saja yang di perlukan oleh sistem dan kebutuhan penunjang penelitian. Berikut kebutuhan-kebutuhan yang diperlukan yaitu kebutuhan sistem, kebutuhan pengguna, kebutuhan fungsional dan juga kebutuhan non-fungsional. Kebutuhan sistem akan digunakan untuk menjelaskan perangkat apa saja yang diperlukan untuk membangun sebuah

sistem. Kebutuhan sistem terbagi menjadi dua yaitu kebutuhan perangkat keras dan kebutuhan perangkat lunak. Kebutuhan perangkat keras akan membahas perangkat keras secara spesifik, dan kebutuhan perangkat lunak membahas tentang perangkat lunak yang dibutuhkan untuk mendukung performa sistem.

Selain itu, ada juga kebutuhan fungsional dan kebutuhan non-fungsional. Pada kebutuhan fungsional akan dijelaskan kegunaan dan manfaat dari sistem yang akan dibuat. Kebutuhan yang dimaksud ialah sistem dapat mendeteksi dan melakukan perhitungan koloni bakteri secara otomatis, kemudian hasil akan ditampilkan pada GUI. Sedangkan kebutuhan non-fungsional akan membahas tentang kebutuhan pengguna, lingkungan operasi, asumsi dan ketergantungan, serta batasan perancangan dan implementasi.

3.4 Perancangan Sistem

Perancangan sistem dilakukan saat analisis kebutuhan sudah selesai dilakukan. Tahap perancangan berfungsi untuk menyelesaikan masalah yang ada pada rumusan masalah. Pada tahap ini akan dilakukan perancangan data sampel, perancangan komunikasi sistem, perancangan algoritma untuk mendeteksi serta perhitungan jumlah koloni bakteri, dan perancangan antarmuka sistem.

3.5 Implementasi Sistem

Tahap implementasi ialah tahap merealisasikan semua rancangan yang sudah dibuat pada tahap sebelumnya. Langkah pertama yang dilakukan ialah mempersiapkan perangkat keras berupa mikroskop dan laptop. Kemudian mempersiapkan perangkat lunak berupa sistem operasi Windows 8.1 Pro dan MATLAB, lalu melakukan deteksi dan perhitungan koloni bakteri *Salmonella*.

3.6 Pengujian dan Analisis

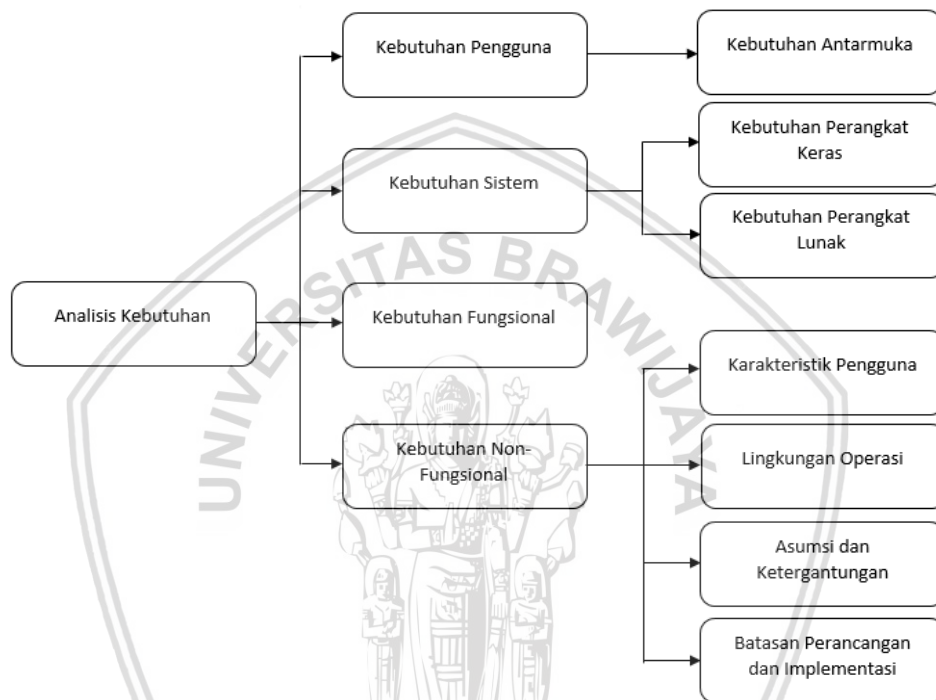
Tahap pengujian dan analisis dilakukan sesuai dengan perancangan sistem. Tujuannya ialah agar mengetahui apakah sistem yang dibuat sudah sesuai dengan yang diharapkan pada tahap perancangan dan implementasi. Hal yang akan dilakukan pada tahap pengujian ialah pengujian kebutuhan fungsional, pengujian akurasi deteksi dan perhitungan koloni bakteri, dan pengujian waktu eksekusi sistem.

3.7 Pengambilan Kesimpulan

Kesimpulan diambil dari hasil pengujian yang telah dilakukan kemudian ditelaah apakah sudah sesuai dengan rumusan masalah. Tujuannya agar didapat hasil penelitian yang mengatasi permasalahan yang ada. Pada tahap ini juga berisi saran yang bertujuan memberikan masukan untuk penelitian selanjutnya agar sistem menjadi lebih baik lagi.

BAB 4 ANALISIS KEBUTUHAN

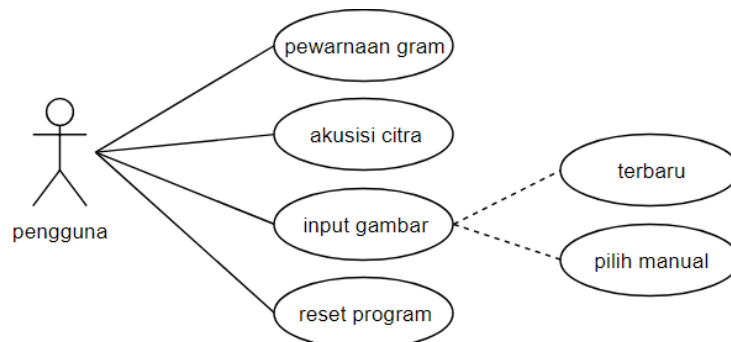
Analisis kebutuhan merupakan definisi dari penguaraian suatu sistem informasi yang utuh kedalam bagian komponen dengan tujuan untuk mengidentifikasi dan mengevaluasi masalah, kesempatan, hambatan, dan kebutuhan sehingga sistem bisa sesuai dengan yang diharapkan. Pada bagian ini akan dijabarkan analisis kebutuhan yang terbagi menjadi empat macam yaitu kebutuhan pengguna, kebutuhan sistem yang terbagi menjadi dua, kebutuhan fungsional, kebutuhan non-fungsional seperti pada Gambar 0.1.



Gambar 0.1 Diagram Analisis Kebutuhan

4.1 Kebutuhan Pengguna

Kebutuhan pengguna merupakan suatu kebutuhan yang harus dimiliki agar memudahkan pengguna dalam monitoring sistem dengan optimal dan didapat hasil yang sesuai dengan kebutuhan. Kebutuhan pengguna pada penelitian ini dapat dilihat pada Gambar 0.2.



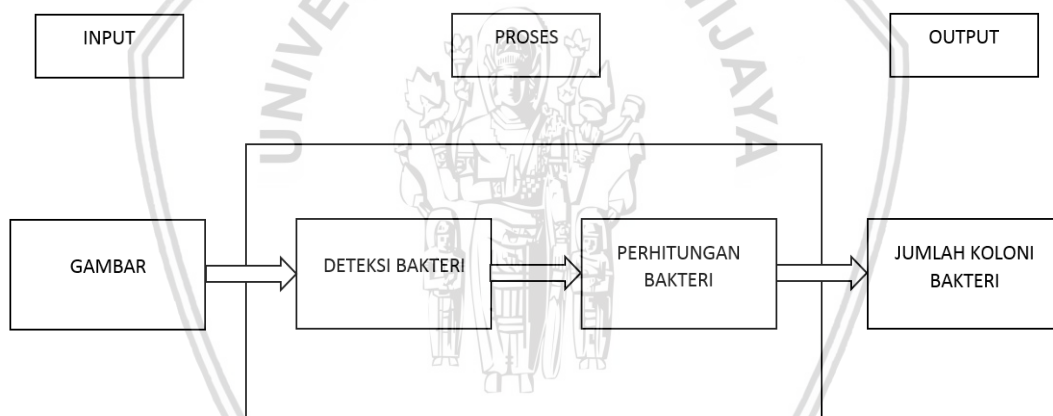
Gambar 0.2 Use Case Kebutuhan Pengguna

4.1.1 Kebutuhan Antarmuka

Kebutuhan antarmuka dengan pengguna berfungsi untuk menampilkan hasil dari citra yang olah serta hasil deteksi dan perhitungan jumlah koloni bakteri. Informasi yang ditampilkan sudah tersusun rapi didalam sebuah GUI, jadi pengguna hanya tinggal memasukan citra sesuai dengan keinginan. Bakteri yang dijadikan sampel sebelumnya sudah melalui tahap pewarnaan gram agar memudahkan saat akan dideteksi dan dilakukan perhitungan jumlah koloni bakteri. Bakteri yang digunakan ialah bakteri *Salmonella* yang masuk ke dalam jenis bakteri gram negatif. Antarmuka yang ditampilkan akan berisi dari hasil pengolahan citra yang dilakukan, hasil deteksi dan perhitungan jumlah koloni bakteri, serta informasi dari gambar yang dijadikan sebagai *input*.

4.2 Kebutuhan Sistem

Sistem pada penelitian ini digunakan untuk mendeteksi dan melakukan perhitungan jumlah koloni bakteri secara otomatis untuk dapat membantu laboran dalam mengidentifikasi dan menghitung jumlah koloni bakteri saat melakukan pengamatan dengan mikroskop digital. Pada Gambar 0.3 menjelaskan gambaran umum sistem.



Gambar 0.3 Diagram Blok Sistem

Gambaran umum sistem seperti yang ada pada Gambar 4.3 terbagi menjadi tiga bagian utama. Bagian pertama ialah *input*, citra yang dimasukan adalah gambar yang sudah diambil oleh mikroskop digital. Bagian selanjutnya ialah bagian proses. Pada bagian ini terdiri dari dua proses yaitu proses deteksi koloni bakteri dan proses perhitungan jumlah koloni bakteri. Proses deteksi koloni bakteri pada citra dilakukan dengan menggunakan segmentasi warna. Sedangkan pada proses perhitungan jumlah koloni bakteri, akan dihitung berapa banyak koloni bakteri yang terdapat dalam citra hasil deteksi dengan menggunakan *object counting*. Setelah semua proses dilakukan, sistem akan menghasilkan output berupa jumlah koloni bakteri. Sistem tersebut akan berjalan didalam sebuah antarmuka untuk mempermudah pengguna.

4.2.1 Kebutuhan Perangkat Keras

Dalam membuat sistem deteksi dan perhitungan jumlah koloni bakteri dibutuhkan beberapa perangkat keras diantaranya :

1. Laptop ASUS X450CP

Laptop ASUS X450CP merupakan laptop dengan memori yang cukup besar untuk menjalankan sebuah sistem operasi dan sebuah software bahasa pemrograman. Maka dari itu Laptop ASUS X450CP mempunyai peran inti dalam pemrosesan sistem deteksi dan perhitungan jumlah koloni bakteri. Gambar 0.4 merupakan bentuk dari laptop ASUS X450CP.



Gambar 0.4 ASUS X450CP

Sumber : (ASUS, 2018)

Dalam penelitian ini menggunakan Laptop ASUS X450CP dengan minimal prosesor core i3 dan 4GB RAM agar pemrosesan sistem dapat berjalan dengan baik. Berikut pada Tabel 0.1 merupakan spesifikasi dari Laptop ASUS X450CP.

Tabel 0.1 Spesifikasi Laptop ASUS X450CP

Nama	Intel® HD Graphics 4000
Manufaktur	Intel Corporation
Tipe Chip	Intel® HD Graphics Family
Sistem Manufaktur	ASUSTek COMPUTER INC.
Sistem Model	X450CP
Prosesor	Intel® Core™ i3-3217U CPU @ 1.80GHz (4 CPUs), ~1.8Ghz
Memori	4096MB RAM

2. Mikroskop Digital Celestron

Celestron LCD Digital *Microscope* II merupakan perangkat keras yang digunakan untuk mengambil gambar ataupun video bakteri saat sedang diamati. Celestron LCD Digital *Microscope* II memiliki kamera dengan kualitas yang cukup baik. Selain itu Celestron LCD Digital *Microscope* II juga memiliki port USB untuk

menyambungkan ke laptop. Pada Gambar 0.5 merupakan bentuk dari Celestron LCD Digital *Microscope* II.



Gambar 0.5 Celestron LCD Digital Microscope II

Sumber : (Celestron, 2018)

Dalam penelitian ini menggunakan Celestron LCD Digital *Microscope* II dengan kamera 5MP dan perbesaran lensa 4x, 10x, dan 40x agar gambar yang dihasilkan dapat terlihat dengan baik. Berikut pada Tabel 0.2 merupakan spesifikasi dari Celestron LCD Digital *Microscope* II.

Tabel 0.2 Spesifikasi Celestron LCD Digital Microscope II

<i>Imaging Sensor</i>	5 MP CMOS $\frac{1}{2}$ 5", 10x magnification in lieu of eyepiece
<i>AC Adapter</i>	Input universal 100 to 240 volt, 50/60 Hz, 4 plug multi-country
<i>USB Connection</i>	2.0, 1.5 m cable included
<i>Magnification (low power)</i>	40x
<i>Magnification (high power)</i>	400 (1600 with digital zoom)x
<i>Continuous Video (VGA)</i>	Yes, 30 fps
<i>Included Accessories</i>	1 GB SD memory card
<i>Objective Nosepiece</i>	3 position with click stop
<i>Objective Lens Diameter</i>	3 achromatic – 4x, 10x., 40x
<i>Condenser Lens</i>	+HEADER+ 0.65
<i>Illumination</i>	Upper and lower, built-in adjustable LEDs
<i>Stage</i>	Fully mechanical with metal clips, 88 mm x 88 mm (3.5" x 3.5")

4.2.2 Kebutuhan Perangkat Lunak

Dalam membuat sistem deteksi dan perhitungan jumlah koloni bakteri dibutuhkan beberapa perangkat lunak untuk mendukung berjalannya sistem, diantaranya :

1. Sistem Operasi Windows 8.1 Pro

Sistem operasi windows merupakan keluarga sistem operasi komputer pribadi yang dikembangkan oleh Microsoft Corporation serta menggunakan antarmuka berbasis GUI. Tidak sulit dalam penggunaannya menjadikan sistem operasi

windows banyak digunakan oleh masyarakat berbagai kalangan. Penelitian ini menggunakan sistem operasi windows versi 8.1 Profesional karena merupakan versi yang stabil untuk menjalankan pemrosesan sistem dan kebutuhan memori sesuai dengan spesifikasi laptop pada kebutuhan perangkat keras.

2. MATLAB R2015a

MATLAB merupakan singkatan dari *Matrix Laboratory* yaitu sebuah program yang bertujuan untuk menganalisis serta mengkomputasi data numerik. Selain itu MATLAB juga dapat digunakan untuk melakukan pemrograman pengolahan citra baik secara *real-time* ataupun tidak. Bahasa pemrograman MATLAB berdasarkan pada baris dan kolom (matriks).

4.3 Kebutuhan Fungsional

Kebutuhan fungsional adalah penguraian sistem yang dibuat dan mendapatkan *output* sesuai keinginan. Hal yang diperlukan dalam bagian ini ialah:

1. Sistem dapat melakukan *input* gambar terakhir yang telah diambil oleh kamera mikroskop ataupun gambar yang dipilih secara manual oleh *user*.
2. Sistem dapat mendeteksi dan melakukan perhitungan otomatis pada koloni bakteri *Salmonella* menggunakan segmentasi warna dan *object counting*.
3. Sistem dapat menampilkan informasi dari *input* yang dipilih serta hasil dari perhitungan jumlah koloni bakteri pada GUI.
4. Sistem dapat melakukan *reset* program jika user sudah selesai ataupun jika user melakukan kesalahan saat *input* gambar.

4.4 Kebutuhan Non-Fungsional

Bab ini akan membahas kebutuhan non-fungsional yang dibutuhkan baik dari segi karakteristik pengguna, lingkungan operasi, asumsi dan ketergantungan, serta batasan perancangan dan implementasi sistem.

4.4.1 Karakteristik Pengguna

Karakteristik pengguna diperuntukkan untuk laboran yang ada di laboratorium mikrobiologi. Dengan adanya sistem ini diharapkan mampu mempermudah laboran dalam mendeteksi koloni bakteri dan menghitung jumlah koloni bakteri saat sedang melakukan pengamatan dengan mikroskop digital.

4.4.2 Lingkungan Operasi

Persyaratan lingkungan operasi untuk menjalankan sistem ialah:

1. Objek yang digunakan adalah bakteri *Salmonella* yang terdapat pada daging ayam.
2. Membutuhkan satu personal computer ataupun laptop yang terhubung dengan kabel USB mikroskop digital agar gambar dapat terlihat secara optimal pada saat menjalankan sistem.
3. Bakteri harus diamati didalam laboratorium dengan menggunakan prosedur yang ada.

4.4.3 Asumsi dan Ketergantungan

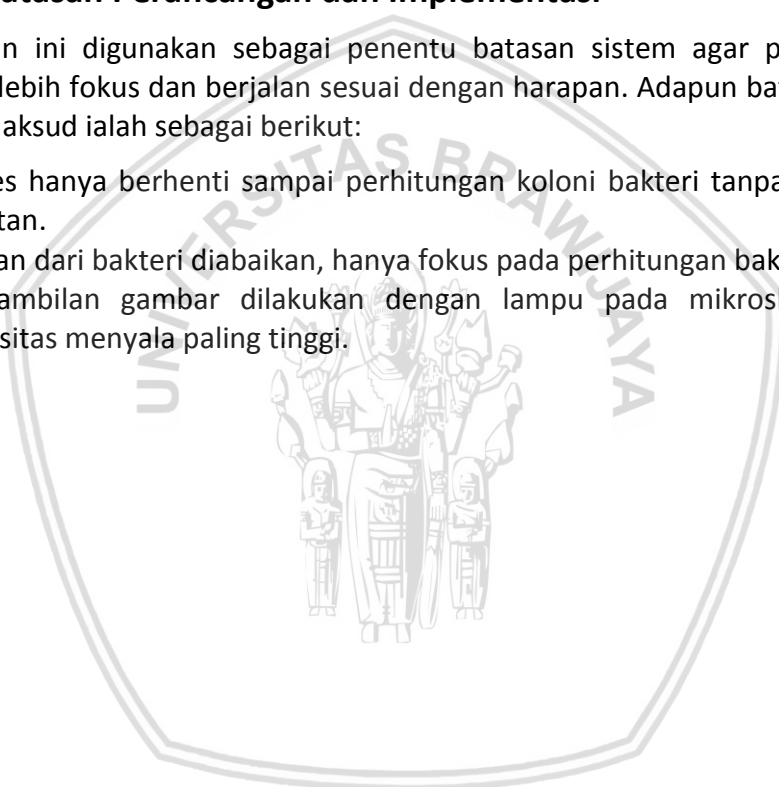
Pada bagian ini, asumsi dan ketergantungan yang akan dibahas agar sistem dapat berjalan sesuai ialah sebagai berikut:

1. Sistem akan mampu mendeteksi dan menghitung jumlah koloni bakteri secara otomatis dengan akurasi yang baik.
2. Bakteri *Salmonella* sudah diberi pewarnaan dengan teknik pewarnaan gram.
3. Proses deteksi dan hitung jumlah koloni bakteri akan berhasil jika perbesaran yang dipakai pada mikroskop adalah 40x.
4. Sistem dapat dijalankan pada komputer ataupun laptop yang minimal memiliki MATLAB versi R2015 dan sistem operasi windows versi 8.1 Profesional.

4.4.4 Batasan Perancangan dan Implementasi

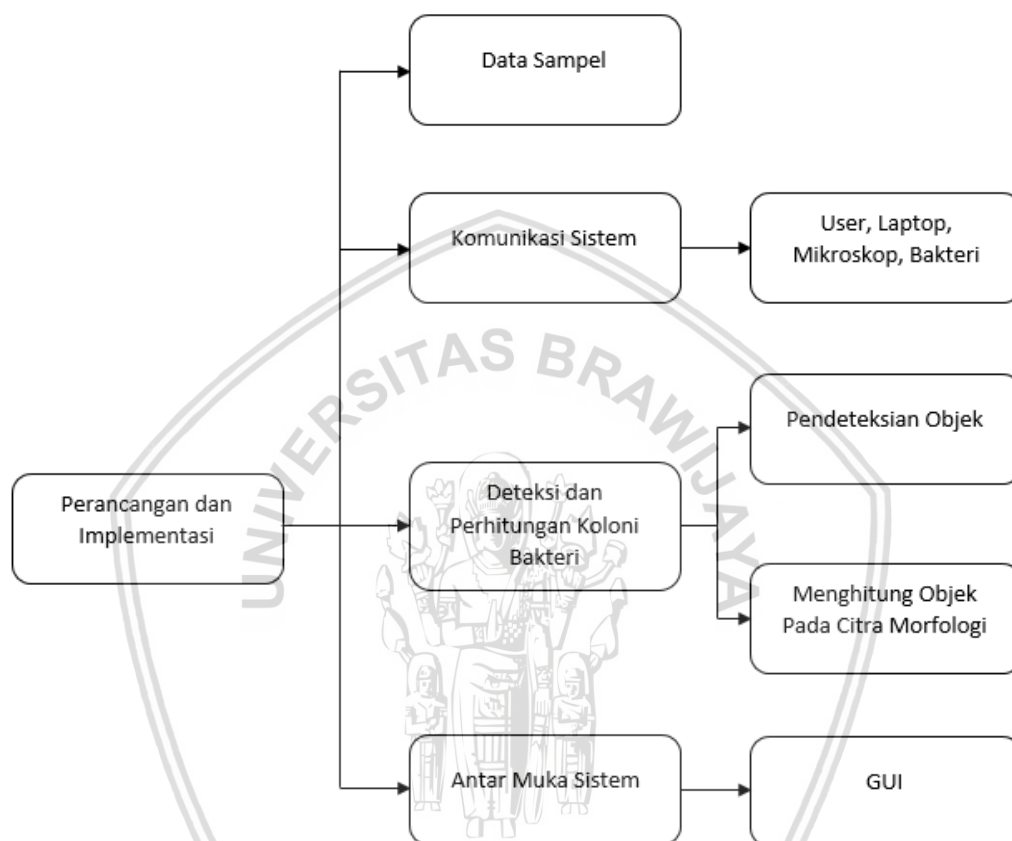
Bagian ini digunakan sebagai penentu batasan sistem agar penelitian ini menjadi lebih fokus dan berjalan sesuai dengan harapan. Adapun batasan sistem yang dimaksud ialah sebagai berikut:

1. Proses hanya berhenti sampai perhitungan koloni bakteri tanpa ada proses lanjutan.
2. Ukuran dari bakteri diabaikan, hanya fokus pada perhitungan bakteri saja.
3. Pengambilan gambar dilakukan dengan lampu pada mikroskop dengan intensitas menyala paling tinggi.



BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab perancangan dan implementasi akan berisi tentang bagaimana sistem dirancang serta diimplementasikan. Pada bab perancangan dan implementasi akan dilakukan beberapa tahap yaitu perancangan data sampel, komunikasi sistem, deteksi dan perhitungan koloni bakteri, serta antarmuka sistem yang dapat dilihat pada Gambar 0.1.



Gambar 0.1 Tahapan Perancangan dan Implementasi

Langkah awal yang dilakukan ialah mempersiapkan data sampel yang akan digunakan. Pada penelitian ini data yang digunakan ialah citra dari bakteri *Salmonella*. Setelah itu memastikan koneksi antara mikroskop dan laptop sudah terkoneksi dengan baik. Setelah mikroskop terkoneksi dengan laptop, *user* akan memasukkan gambar sesuai dengan yang diinginkan, baik gambar yang dipilih secara manual ataupun gambar yang ditentukan oleh sistem. Setelah gambar berhasil melakukan proses memasukkan gambar, akan dilakukan proses *image enhancement* dengan metode operasi spasial yaitu *high pass filter* untuk menajamkan tepi objek dan mengurangi *blur*. Kemudian berlanjut ke proses segmentasi warna dengan mengubah citra RGB ke citra HSV untuk diambil nilai saturasinya dan selanjutnya dilakukan proses mengubah citra saturasi ke dalam citra biner dengan *thresholding* untuk menentukan objek tersebut koloni bakteri atau bukan. Setelah didapatkan citra biner, selanjutnya dilakukan *filling holes* dan *area opening* untuk mengisi lubang pada objek. Langkah selanjutnya ialah

melakukan operasi morfologi dengan metode *closing* untuk menyambungkan koloni bakteri yang berdekatan dan mengurangi *noise*. Jika citra hasil morfologi sudah didapatkan, selanjutnya melakukan perhitungan jumlah koloni bakteri yang ada pada citra tersebut. Proses pengolahan citra dan hasil perhitungan jumlah koloni akan ditampilkan dalam sebuah antarmuka sistem.

5.1 Data Sampel

5.1.1 Perancangan Data Sampel

Perancangan data sampel dilakukan dengan membuat preparat dari koloni bakteri *Salmonella*. Pembuatan preparat bakteri membutuhkan koloni bakteri, kaca preparat, serta larutan pewarnaan gram agar koloni bakteri dapat dilihat pada mikroskop digital. Pembuatan preparat bakteri harus dilakukan pada laminar *air flow* agar bakteri tidak menyebar dan udara tidak terkontaminasi oleh bakteri. Pembuatan preparat harus mengikuti prosedur yang sudah ditetapkan pada laboratorium dan harus berhati-hati. Setelah preparat bakteri berhasil dibuat, ditambahkan minyak imersi pada saat akan melakukan pengamatan dengan mikroskop digital. Minyak imersi berguna untuk memperjelas objek dan melindungi lensa perbesaran mikroskop agar tidak terkontaminasi oleh bakteri. Saat melakukan pengamatan dengan mikroskop digital, *user* dapat mengakuisisi citra bakteri yang akan dideteksi serta dihitung. Pengamatan dengan mikroskop digital menggunakan perbesaran lensa objektif 40x agar objek dapat terlihat dengan jelas.

5.1.2 Implementasi Data Sampel

Pembuatan serta pengambilan data sampel akan melalui beberapa tahap. Pada Gambar 0.2 dan Gambar 0.3 merupakan pembuatan preparat bakteri yang dilakukan pada laminar *air flow*.



Gambar 0.2 Laminar Air Flow



Gambar 0.3 Biakan Bakteri Salmonella dan Larutan Pewarnaan Gram

Setelah preparat bakteri berhasil dibuat, kemudian preparat yang sudah diberi minyak imersi tersebut diamati dengan mikroskop digital menggunakan perbesaran lensa objektif 40x. Pada penelitian ini menggunakan perbesaran lensa objektif sebesar 40x agar gambar dari bakteri dapat terlihat dengan jelas. Pada Gambar 0.4 merupakan perbandingan gambar bakteri dengan perbesaran lensa objektif 4x, 10x, dan 40x.



Gambar 0.4 Mikroskop Dengan Perbesaran Lensa Objektif 4x, 10x, dan 40x

5.2 Komunikasi Sistem

5.2.1 Perancangan Komunikasi Sistem

Pada perancangan sistem ini, agar memudahkan pemahaman terhadap sistem secara keseluruhan maka dibuat diagram blok yang dapat dilihat pada Gambar 0.5.



Gambar 0.5 Diagram Blok Komunikasi Sistem

Pada perancangan komunikasi sistem ini, pertama kali *user* melakukan pengamatan pada bakteri *Salmonella* menggunakan mikroskop digital. Kemudian setelah itu *user* mengambil gambar bakteri *Salmonella* tersebut. Setelah didapat gambar yang diinginkan, maka *user* menyambungkan kabel USB agar laptop dapat membaca gambar dari mikroskop. Lalu setelah laptop dan mikroskop terkoneksi dengan baik, *user* membuka antarmuka sistem. Setelah antarmuka sistem tersebut dijalankan, *user* akan diberikan pilihan untuk memasukkan gambar yang paling terakhir diambil oleh mikroskop ataupun memilih gambar secara manual.

Jika *user* sudah memilih, proses akan berjalan dan akan menampilkan hasil yang dibutuhkan oleh *user*.

5.2.2 Implementasi Komunikasi Sistem

Komunikasi yang terjadi antara mikroskop digital dan laptop ialah komunikasi serial. Komunikasi antara mikroskop digital dan laptop hanya memerlukan satu jalur atau kabel yang sedikit daripada komunikasi paralel. Komunikasi antara mikroskop digital dengan laptop hanya support kabel USB saja. Setelah terlihat bakteri pada mikroskop, *user* mengambil gambar bakteri tersebut dengan mikroskop digital. Kemudian menyambungkan kabel USB mikroskop ke laptop seperti Gambar 0.6.



Gambar 0.6 Menyambungkan Mikroskop ke Laptop Dengan Kabel USB

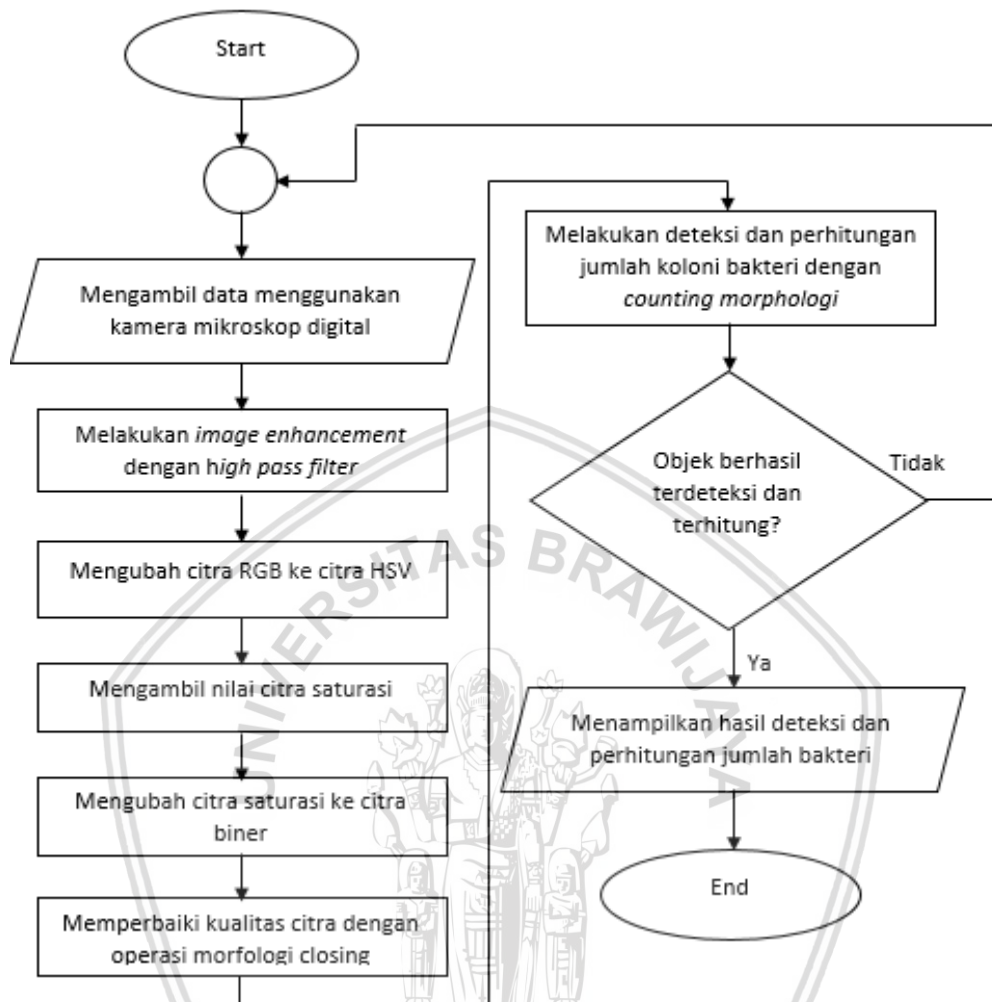
Pada saat laptop dan mikroskop sudah tersambung dengan baik, *user* akan menjalankan antarmuka sistem deteksi dan perhitungan koloni bakteri. Kemudian *user* akan memilih masukan gambar yang digunakan. Ada dua jenis masukan yang disediakan yaitu masukan gambar terakhir yang diambil oleh mikroskop dan masukan gambar yang dipilih secara manual oleh *user*.

5.3 Deteksi dan Perhitungan Koloni Bakteri

5.3.1 Perancangan Deteksi dan Perhitungan Koloni Bakteri

Dalam perancangan deteksi dan perhitungan objek, yang diperlukan sebagai masukan adalah data dari kamera pada mikroskop digital. Setelah data pada kamera dimasukkan oleh *user* maka langkah selanjutnya ialah proses deteksi dan perhitungan objek dengan pengolahan citra. Metode yang digunakan diantaranya ialah meningkatkan kualitas citra dengan salah satu teknik *image enhancement*, yaitu *high pass filter* agar citra koloni bakteri lebih mudah untuk dideteksi. Setelah itu tahapan selanjutnya ialah mengkonversi nilai citra RGB ke citra HSV untuk diambil nilai citra saturasinya. Kemudian citra saturasi tersebut diubah ke citra biner agar dapat dilakukan operasi morfologi guna menghasilkan citra sesuai dengan kebutuhan. Setelah dilakukan operasi morfologi, kemudian dilakukan operasi perhitungan jumlah koloni bakteri dengan *object counting* untuk mengetahui berapa banyak jumlah koloni bakteri yang ada pada citra tersebut.

Gambar 0.7 merupakan alur perancangan dari pendeteksian dan perhitungan koloni bakteri.



Gambar 0.7 Alur Pendeteksian dan Perhitungan Jumlah Koloni bakteri

5.3.1.1 Melakukan *Image Enhancement* Dengan *High Pass Filter* Pada Citra RGB

Langkah pertama yang dilakukan setelah *user* melakukan *input* gambar ialah, melakukan *image enhancement* untuk mempertajam tepi objek dan menghilangkan *blur* yang dihasilkan saat pengambilan gambar. Persamaan 2.2 sampai persamaan 2.4 merupakan rumus yang dapat digunakan untuk melakukan teknik *high pass filter* dengan kernel tiga. Perhitungannya ialah sebagai berikut:

Sebuah gambar mempunyai nilai matrik

$$X = \begin{bmatrix} 10 & 20 & 30 \\ 10 & 50 & 30 \\ 10 & 20 & 30 \end{bmatrix}$$

Pada gambar tersebut akan dilakukan metode *high pass filter* menggunakan kernel tiga yang memiliki nilai matrik

$$H3 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix}$$

Setelah nilai dari kedua matrik diketahui, maka dilakukan proses konvolusi atau nama lain dari perhitungan operasi spasial yang terdapat pada persamaan 2.2. Langkah pertama ialah mencari nilai dari koordinat (2,2) dengan melakukan operasi perkalian pada matrik. Sebelumnya, matrik dari kernel H3 harus diputar sebesar 180° agar dapat dilakukan operasi perkalian.

$$y(2,2) = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix} * \begin{bmatrix} 10 & 20 & 30 \\ 10 & 50 & 30 \\ 10 & 20 & 30 \end{bmatrix}$$

$$y(2,2) = 10 + (-40) + 30 + (-20) + 250 + (-60) + 10 + (-40) + 30$$

$$y(2,2) = 330 - 160 = 170$$

Setelah didapatkan nilai hasil *high pass filter* dari koordinat (2,2) selanjutnya mencari nilai dari semua koordinat yang tersisa dengan cara yang sama. Hanya saja menggunakan metode *zero padding* atau dengan kata lain menambahkan nilai nol seperti pada Gambar 0.8 sehingga membentuk sebuah matrik, agar nilai dari koordinat yang dicari dapat dikalikan dengan nilai pada bagian tengah.



Gambar 0.8 Metode Zero Padding

Maka perhitungan pada koordinat (1,1) ialah

$$y(1,1) = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix} * \begin{bmatrix} 0 & 0 & 0 \\ 0 & 10 & 20 \\ 0 & 10 & 50 \end{bmatrix}$$

$$y(1,1) = 0 + 0 + 0 + 0 + 50 + (-40) + 0 + (-20) + 50$$

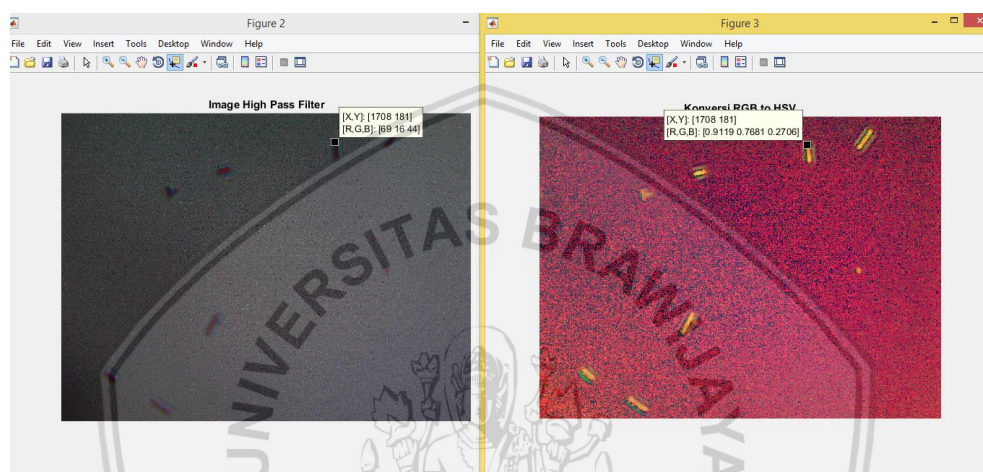
$$y(1,1) = 100 - 60 = 40$$

Cara yang sama juga digunakan untuk mencari nilai koordinat yang lainnya sehingga akan didapatkan hasil akhir dari penggunaan *high pass filter* ialah sebagai berikut :

$$O = \begin{bmatrix} 40 & 0 & 100 \\ 0 & 170 & 0 \\ 40 & 0 & 100 \end{bmatrix}$$

5.3.1.2 Mengubah Citra RGB ke Citra HSV

Setelah melakukan *image enhancement* dengan menggunakan *high pass filter*, langkah selanjutnya ialah mengubah citra rgb menjadi citra. Untuk melakukan konversi gambar dari rgb ke hsv digunakan fungsi pada MATLAB yang mempunyai parameter citra hasil dari *image enhancement*. Terlihat pada Gambar 0.9 di *figure 2* dan *figure 3* di koordinat x,y terjadi perubahan nilai RGB yang sebelumnya adalah 69,16,44 menjadi 0.9119, 0.7681, 0.2706 pada citra HSV.



Gambar 0.9 Letak Koordinat X,Y Pada Citra

Untuk mengkonversi nilai RGB hasil *image enhancement* digunakan rumus seperti pada persamaan 2.5 sampai persamaan 2.11 dengan perhitungan sebagai berikut :

$$R' = 69/255 = 0.2705$$

$$G' = 16/255 = 0.0627$$

$$B' = 44/255 = 0.1725$$

Setelah didapatkan nilai RGB aksien, dicari nilai masing-masing dari HSV

$$V = \max(R', G', B') = 0.2705$$

$$\Delta = V - \min(R', G', B') = 0.2705 - \min(0.2705, 0.0627, 0.1725)$$

$$\Delta = 0.2705 - 0.0627 = 0.2078$$

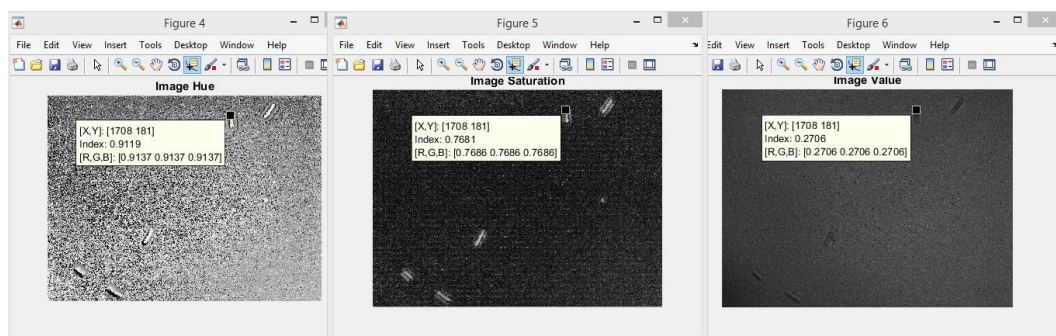
$$S = \Delta/V = 0.2078/0.2705 = 0.7682$$

$$H = 60^\circ \times ((G' - B')/\Delta) \bmod 6$$

$$H = 60^\circ \times ((-0.1098)/0.2078) \bmod 6$$

$$H = 0.9119$$

Maka hasil yang didapat akan sesuai dengan nilai yang ada pada MATLAB seperti pada Gambar 0.10, dimana nilai *hue* sebesar 0.9119, nilai saturasi sebesar 0.7681, dan nilai *value* sebesar 0.2706.



Gambar 0.10 Hasil Nilai HSV dari koordinat yang sama

Rumus tersebut juga berlaku untuk mencari nilai HSV pada titik koordinat lainnya.

5.3.1.3 Mengubah Citra Saturasi ke Citra Biner

Mengubah citra saturasi ke dalam citra biner dilakukan dengan menggunakan metode *thresholding*. Langkah awalnya ialah menggunakan fungsi `im2bw` pada MATLAB dengan parameter *input* citra saturasi serta menetapkan nilai *threshold* yang digunakan. Persamaan 2.1 merupakan rumus yang dapat digunakan untuk mengubah atau mengkonversi suatu gambar ke biner. Perhitungannya sebagai berikut:

Sebagai contoh, misalnya pada suatu citra di dapatkan nilai matriks

$$I = \begin{bmatrix} 40 & 160 & 69 \\ 20 & 250 & 140 \\ 70 & 30 & 128 \end{bmatrix}$$

Nilai $40 < 128$, maka nilainya diubah menjadi 0. Sedangkan $160 > 128$, maka nilainya diubah menjadi 255. Begitu juga dengan nilai matrik yang lain. Jika lebih kecil dari 128 maka diubah menjadi 0, tapi jika lebih besar dari 128 maka diubah menjadi 1. Proses ini akan terus dilakukan dalam sistem ini. Maka hasil dari gambar yang telah diubah ke biner memiliki nilai matrik sebagai berikut :

$$I = \begin{bmatrix} 0 & 255 & 0 \\ 0 & 255 & 255 \\ 0 & 0 & 255 \end{bmatrix}$$

5.3.1.4 Memperbaiki Kualitas Citra Dengan Operasi Morfologi

Setelah citra saturasi dikonversi kedalam citra biner, selanjutnya ialah menghilangkan *noise* pada citra biner tersebut dengan operasi morfologi *closing*. Untuk melakukan operasi morfologi *closing* langkah awal yang dilakukan ialah pengisian lubang-lubang pada gambar dengan fungsi `imfill` dan `bwareaopen` pada MATLAB yang memiliki *input* parameter citra biner. Untuk melakukan operasi morfologi *closing* ada beberapa tahap yang harus dilalui yaitu :

1. Menentukan *structure element* apa yang akan digunakan seperti yang sudah dijelaskan pada dasar teori. Pada penelitian ini *structure element* yang digunakan ialah bentuk *square* dengan radius 9x9.
2. Melakukan operasi morfologi dilasi terlebih dahulu seperti rumus yang ada pada persamaan 2.12.
3. Kemudian, melakukan operasi morfologi erosi seperti rumus yang ada pada persamaan 2.13 dengan menggunakan parameter *input* citra hasil morfologi dilasi dan *structure element* yang digunakan.
4. Pada tahap akhir, dilakukan operasi morfologi closing yaitu dengan parameter *input* citra hasil morfologi dilasi dan *structure element*.

5.3.1.5 Melakukan Deteksi dan Perhitungan Jumlah Koloni Bakteri Dengan *Object counting*

Untuk membedakan koloni bakteri dengan *noise* dan mengetahui berapa jumlah koloni bakteri yang terdapat pada citra maka diperlukan pendeteksian serta perhitungan pada masukan citra yang sudah diolah sebelumnya. Untuk melakukan pendeteksian dan perhitungan ini digunakan fungsi *regionprops* pada MATLAB yang berfungsi untuk mengukur sekumpulan properti dari setiap *region* dan juga menggunakan *centroid* untuk menentukan titik tengah suatu objek. Jika koloni bakteri berhasil dideteksi maka sistem akan menampilkan hasil deteksi dan perhitungan tersebut yang sudah diberi *boundingbox*.

5.3.2 Implementasi Deteksi dan Perhitungan Koloni Bakteri

Pendeteksian dan perhitungan jumlah koloni bakteri pada sistem ini akan melalui beberapa tahapan. Citra yang menjadi *input* sistem akan dinaikan kualitas gambarnya menggunakan *image enhancement* dengan metode *high pass filter*. Kemudian hasil *image enhancement* dikonversi atau di segmentasi ke dalam ruang warna HSV. Selanjutnya, citra saturasi dikonversi ke dalam citra biner untuk mempermudah dalam melakukan operasi morfologi *closing*. Langkah selanjutnya ialah melakukan operasi morfologi untuk memperhalus dan menghilangkan *noise* pada citra biner yang dijadikan sebagai *input* parameter. Deteksi dan perhitungan baru bisa dilakukan saat proses morfologi *closing* selesai dilakukan. Deteksi dan perhitungan jumlah koloni bakteri ini menggunakan metode *object counting*.

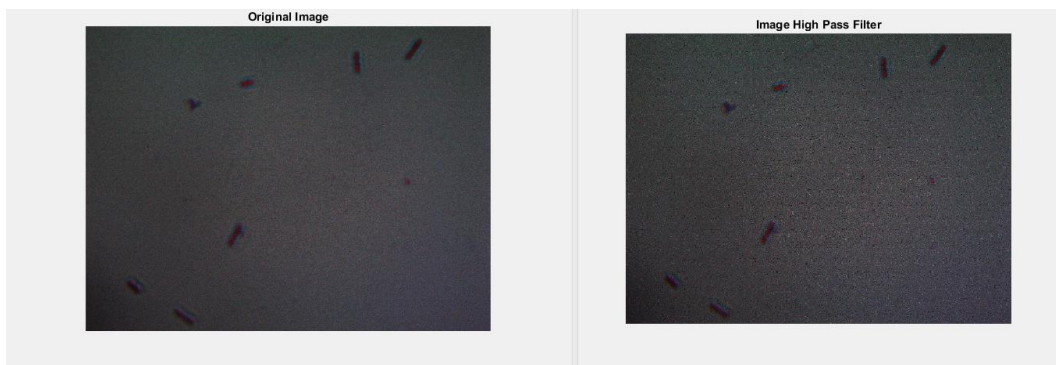
5.3.2.1 Melakukan *Image Enhancement* Dengan *High Pass Filter*

Input dari tahapan ini ialah citra yang dipilih oleh user. Untuk melakukan *image enhancement* digunakan metode *high pass filter* yang dapat dilihat pada Tabel 0.1.

Tabel 0.1 Implementasi *high pass filter*

<i>Image enhancement dengan high pass filter</i>	
1	hpf3=[1 -2 1;-2 5 -2; 1 -2 1];
2	K3=uint8(convn(double(A),hpf3,'same'));

Implementasi dari *image enhancement* dengan metode *high pass filter* kernel ketiga dapat dilihat pada Gambar 0.11.



Gambar 0.11 Implementasi Image Enhancement

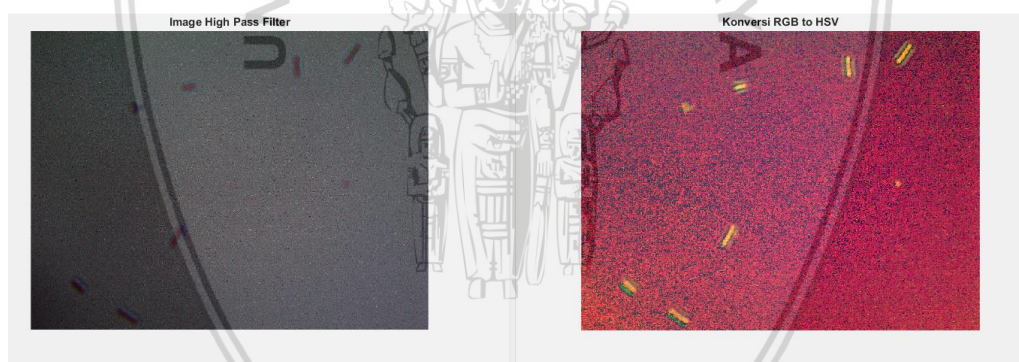
5.3.2.2 Mengubah Citra RGB ke Citra HSV ke Citra HSV

Input pada tahapan ini menggunakan citra hasil *image enhancement*. Implementasi dari tahap ini dapat dilihat pada Tabel 0.2.

Tabel 0.2 Konversi Citra RGB ke Citra HSV

Mengubah pixel gambar dari RGB ke HSV	
1	<code>hsv = rgb2hsv(K3);</code>

Implementasi dari konversi gambar dari RGB ke HSV dapat dilihat pada Gambar 0.12.



Gambar 0.12 Konversi Citra RGB ke Citra HSV

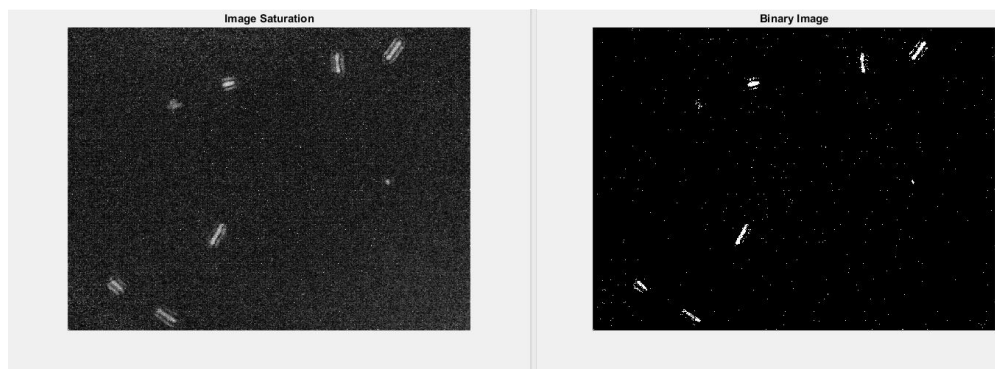
5.3.2.3 Mengubah Citra Saturasi ke Citra Biner

Langkah selanjutnya ialah mengambil nilai citra saturasi untuk diubah ke citra biner. *Input* dari tahap ini ialah citra saturasi seperti yang dapat dilihat pada Tabel 0.3.

Tabel 0.3 Konversi Citra Saturasi ke Citra Biner

Mengubah Citra Saturasi ke Citra Biner	
1	<code>bw = im2bw(S, .5);</code>

Implementasi dari tahap ini dapat dilihat pada Gambar 0.13.



Gambar 0.13 Konversi citra saturasi ke citra biner

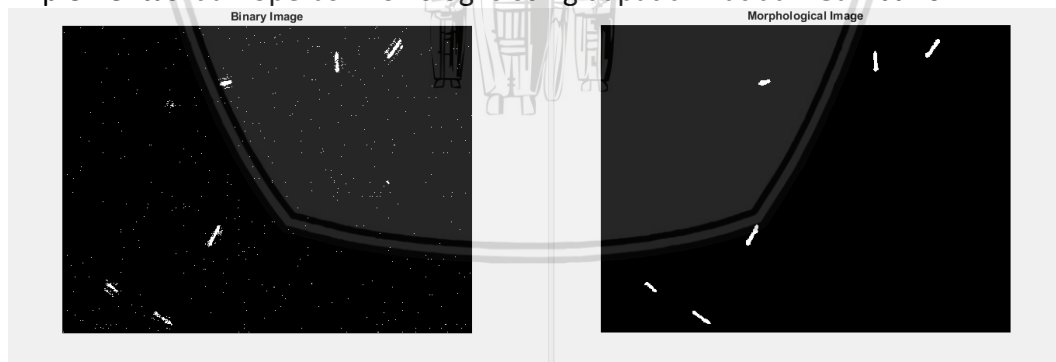
5.3.2.4 Memperbaiki Kualitas Citra Dengan Operasi Morfologi

Input yang digunakan ialah citra biner dengan menggunakan *structure element* bentuk *square* dengan radius 9x9. Sebelum operasi morfologi *closing* dilakukan terlebih dahulu menghilangkan lubang-lubang pada citra biner seperti yang terdapat pada Tabel 0.4.

Tabel 0.4 Operasi Morfologi Closing

Memperbaiki Kualitas Citra Dengan Operasi Morfologi Closing	
1	<code>bw1 = imfill(bw, 'holes');</code>
2	<code>bw2 = bwareaopen(bw1, 1000);</code>
3	<code>str = strel('square', 9);</code>
4	<code>dil = imdilate(bw2, str);</code>
5	<code>er = imerode(dil, str);</code>
6	<code>bw3 = imclose(er, str);</code>

Implementasi dari operasi morfologi closing dapat dilihat dari Gambar 0.14.



Gambar 0.14 Operasi Morfologi Closing

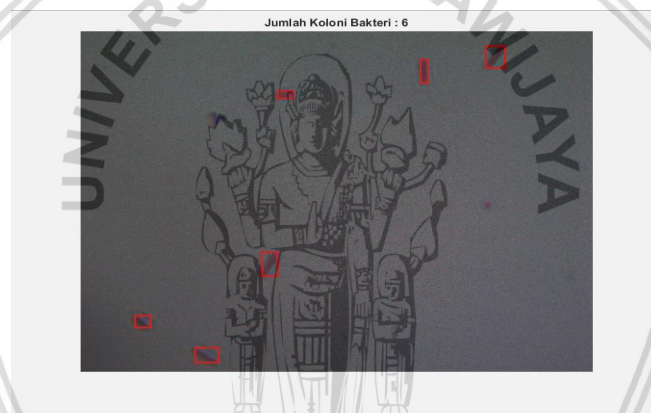
5.3.2.5 Melakukan Deteksi dan Perhitungan Jumlah Koloni Bakteri Dengan *Object counting*

Input yang digunakan pada proses ini ialah citra hasil morfologi. Menggunakan fungsi *regionprops* agar dapat menentukan dan menghitung jumlah koloni bakteri pada gambar yang diolah seperti yang dapat dilihat pada Tabel 0.5.

Tabel 0.5 Deteksi dan Perhitungan Jumlah Koloni bakteri

Deteksi dan Perhitungan Jumlah Koloni bakteri	
1	<code>region = regionprops(bw3, 'centroid');</code>
2	<code>[labeled,numObjects] = bwlabel(bw3,8);</code>
3	<code>statusreg =</code>
4	<code>regionprops(labeled,'Eccentricity','Area','BoundingBox');</code>
5	<code>eccentricities = [statusreg.Eccentricity];</code>
6	<code>ecc1 = find(eccentricities);</code>
7	<code>statsDefects = statusreg(ecc1);</code>

Deteksi dan perhitungan jumlah koloni bakteri menggunakan citra hasil morfologi *closing*, kemudian dicari *centroid* atau titik tengah dari citra bakteri tersebut. Setelah itu area yang sudah diberi label dan sudah ditemukan titik tengahnya akan diberikan *boundingbox* untuk menandai objek bakteri tersebut. Pada Gambar 0.15 dapat dilihat bahwa sistem mampu mendeteksi dan menghitung jumlah koloni bakteri dengan menggunakan metode *object counting*. Koloni bakteri yang terdeteksi akan diberikan tanda dengan kotak berwarna merah atau *boundingbox* dan nanti akan dihitung ada berapa *boundingbox* yang ada didalam gambar tersebut.

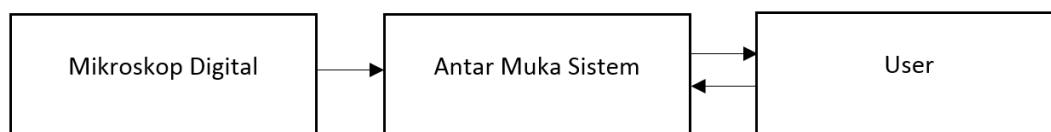


Gambar 0.15 Implementasi Deteksi dan Perhitungan Jumlah Koloni bakteri

5.4 Antarmuka Sistem

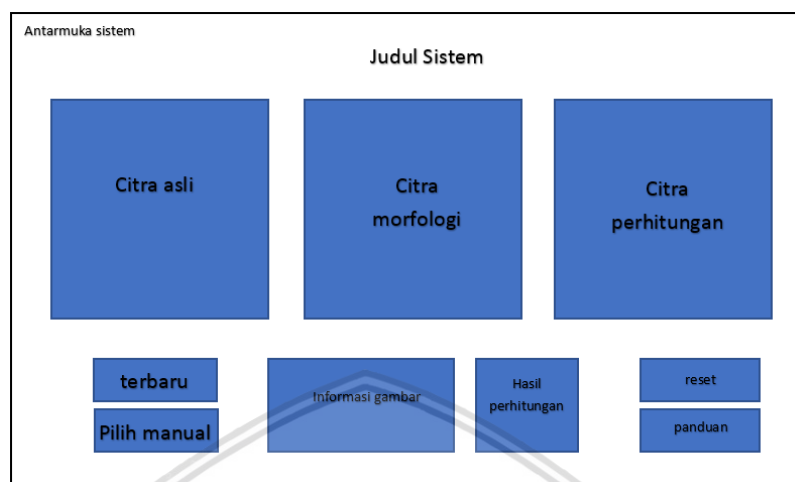
5.4.1 Perancangan Antarmuka Sistem

Pada tahap ini akan dibuat sebuah antarmuka sistem guna memudahkan *user* dalam menjalankan sistem deteksi dan perhitungan koloni bakteri. Antarmuka sistem dibuat dengan menggunakan *library* yang ada pada MATLAB. Pada Gambar 0.16 merupakan diagram blok dari antarmuka sistem yang akan dibuat.



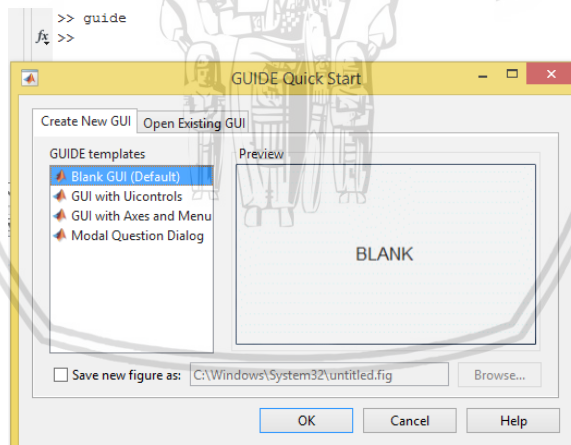
Gambar 0.16 Diagram Blok Perancangan Antarmuka Sistem

Antarmuka sistem dibuat dengan memuat semua proses pengolahan citra yang ada pada bagian perancangan dan implementasi sistem. Pada Gambar 0.17 merupakan desain dari antarmuka sistem yang akan dibuat.



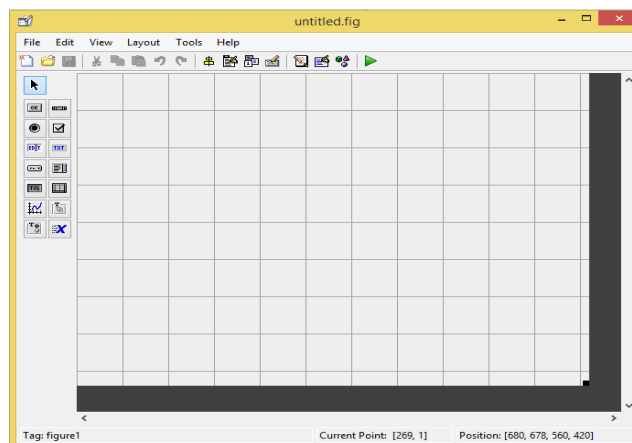
Gambar 0.17 Desain Antarmuka Sistem

Ada tiga kotak yang akan berisi proses pengolahan citra yang dilakukan, empat *pushbutton* untuk membantu *user* saat mengeksekusi sistem, serta kotak hasil perhitungan dan informasi gambar yang dimasukkan. Tahap awal dalam membuat sebuah antarmuka ialah menjalankan *software* MATLAB R2015a, kemudian ketik “*guide*” pada *command window* seperti pada Gambar 0.18.



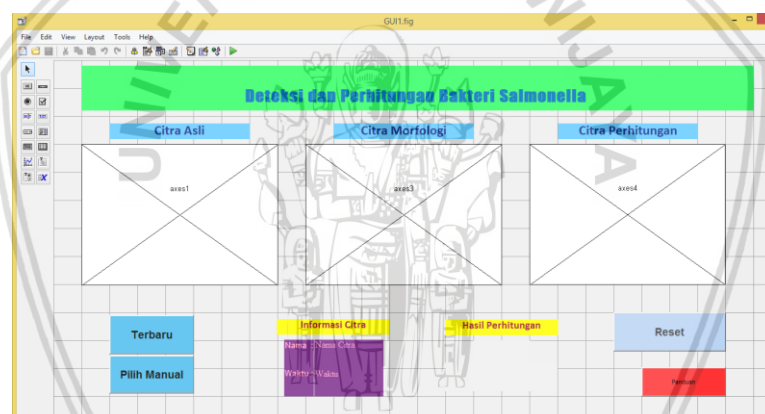
Gambar 0.18 Membuat GUI Pada MATLAB

Setelah memilih *template* dari GUI yang ingin dibuat, maka akan keluar sebuah kotak dialog seperti pada Gambar 0.19.



Gambar 0.19 Tampilan Awal Pembuatan GUI Pada MATLAB

Langkah selanjutnya ialah mendesain GUI sesuai dengan keinginan. Pembuatan desain GUI pada MATLAB cukup sederhana yaitu dengan *drag and drop* pilihan property yang tersedia disisi sebelah kiri lembar kerja. Dengan menerapkan beberapa prosedur yang baik untuk interaksi manusia dan komputer maka dibuat GUI dengan desain seperti Gambar 0.20.

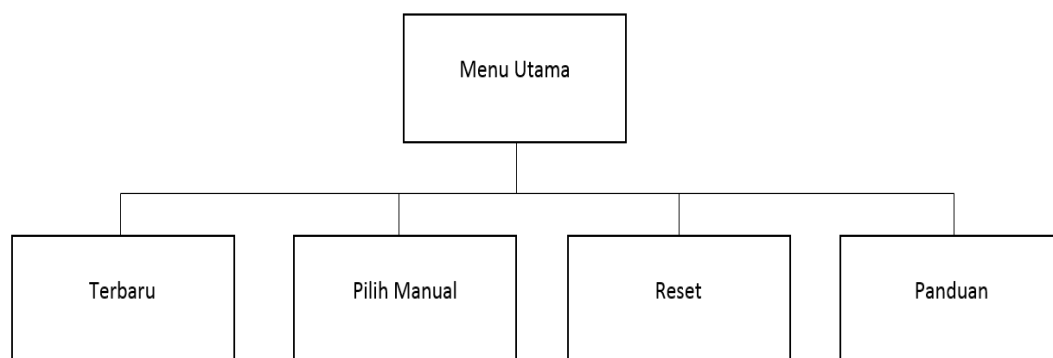


Gambar 0.20 Tampilan Desain GUI

Pada desain GUI sistem deteksi dan perhitungan jumlah koloni bakteri terdapat tiga properti *axes* yang berguna untuk menampilkan citra masukan, citra hasil morfologi, dan citra dengan hasil deteksi dan perhitungan jumlah koloni bakteri. Terdapat empat *pushbotton* yang terdiri dari *pushbotton* terbaru, pilih manual, *reset*, panduan.

5.4.1.1 Menu Utama Antarmuka Sistem

Antarmuka sistem ini terdiri dari satu menu utama yang dapat dilihat pada Gambar 0.21.



Gambar 0.21 Diagram Blok Perancangan Menu Utama Antarmuka Sistem

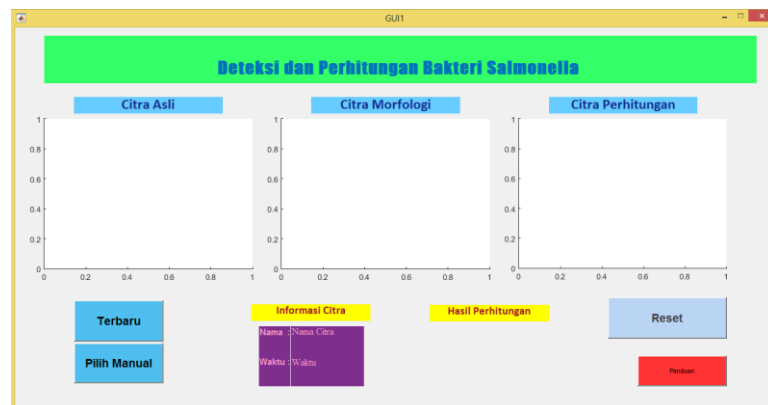
Pada Tabel 0.6 adalah penjelasan untuk masing-masing sub menu yang ada pada menu utama antarmuka sistem yang akan dibuat.

Tabel 0.6 Penjelasan Sub Menu Pada Menu Utama

Sub Menu	Penjelasan
Terbaru	Merupakan pilihan dengan <i>input</i> gambar terakhir yang berhasil di <i>capture</i> oleh kamera pada mikroskop digital.
Pilih Manual	Merupakan pilihan dengan <i>input</i> manual atau <i>user</i> memilih sendiri gambar yang akan diinputkan
Reset	Melakukan set ulang sistem dengan <i>input</i> yang sudah dijalankan agar bisa menjalankan sistem dengan <i>input</i> yang baru.
Panduan	Berupa panduan menggunakan sistem deteksi dan perhitungan jumlah koloni bakteri.

5.4.2 Implementasi Antarmuka Sistem

Implementasi antarmuka sistem dibuat setelah perancangan telah dibuat, langkah selanjutnya ialah membuat *source code* setiap sub menu untuk antarmuka sistem yang akan di implementasikan. Pada Gambar 0.22 merupakan implementasi dari desain GUI yang sudah dibuat.



Gambar 0.22 Implementasi GUI

1. Terbaru

Saat *user* memilih untuk memilih masukan gambar terbaru, maka sistem akan melakukan *sorting* berdasarkan tanggal dan waktu terkini yang ada pada gambar di mikroskop digital seperti pada Tabel 0.7 berikut.

Tabel 0.7 Input Gambar Terbaru

Input terbaru	
1	folder=uigetdir('D:\DCIM\100DSCIM\','Pilih Folder');
2	isi=dir(strcat(folder,\'*.jpg\'));
3	for i=1:length(isi);
4	tanggal(i,1)=i;
5	tanggal(i,2)=isi(i).datetime;
6	end
7	tanggal=sortrows(tanggal,2);
8	namafile=strcat(folder,\'\',isi(tanggal(end,1)).name);

2. Pilih Manual

Ketika *user* memilih menginputkan gambar secara manual, maka sistem akan menampilkan folder *image* yang tersedia pada laptop yang digunakan yang dapat dilihat pada Tabel 0.8 berikut.

Tabel 0.8 Input Gambar Pilih Manual

Input Secara Manual	
1	[nama_file1, nama_path1]=uigetfile(...
2	{\'*.bpm;*.jpg\',\'File Citra(*.bpm,*.jpg)\';
3	\'*.bpm\',\'File Bitmap(*.bpm)\';...
4	\'*.jpg\',\'File jpeg(*.jpg)\';
5	\'*.*\',\'Semua File(*.*)\'},...
6	\'Buka File Citra Asli\');
7	if ~isequal(nama_file1, 0)
8	handles.data1=imread(fullfile(nama_path1,nama_file1));
9	guidata(hObject,handles);
10	handles.current_data1=handles.data1;
11	axes(handles.axes1);
12	imshow(handles.current_data1);
13	else
14	return;
15	end

3. *Reset*

Saat sistem sudah selesai mendeteksi dan melakukan perhitungan jumlah koloni bakteri maka *user* dapat melakukan set ulang antarmuka sistem agar dapat memasukan gambar lain yang akan di deteksi dan di hitung jumlah koloni bakterinya seperti pada yang terlihat di Tabel 0.9.

Tabel 0.9 Reset Sistem

<i>Reset</i>	
1	<code>I=0;</code>
2	<code>plot(handles.axes1,I)</code>
3	<code>newplot</code>

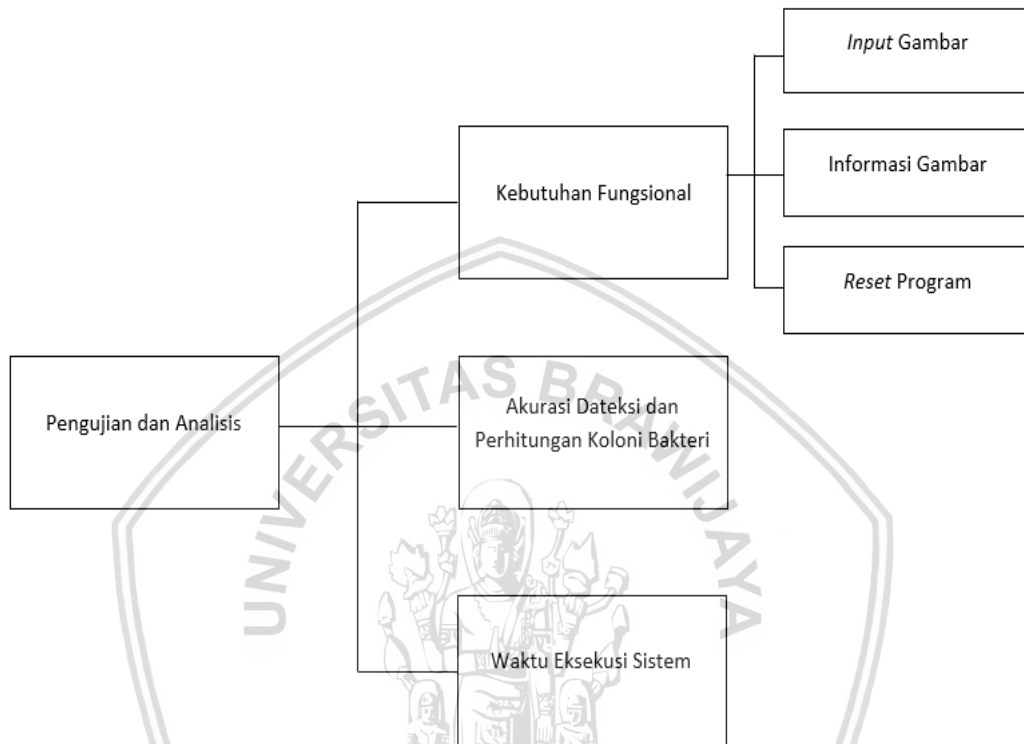
4. Panduan

Sub menu ini merupakan penjelasan untuk *user* mengenai tata cara penggunaan antarmuka sistem.



BAB 6 PENGUJIAN DAN ANALISIS

Bab pengujian dan analisis membahas tentang proses pengujian yang dilakukan pada sistem deteksi dan perhitungan jumlah koloni bakteri serta analisis yang dibuat berdasarkan hasil pengujian yang ada. Dalam sistem ini ada beberapa pengujian yang dilakukan, seperti yang ditunjukkan pada Gambar 0.1.



Gambar 0.1 Diagram Blok Pengujian Sistem

6.1 Pengujian Kebutuhan Fungsional

Pengujian kebutuhan fungsional yang dilakukan ialah pengujian saat *input* gambar, pengujian informasi gambar, dan pengujian *reset* program.

6.1.1 Pengujian *Input* Gambar

6.1.1.1 Tujuan Pengujian

Tujuan dilakukannya pengujian ini ialah untuk menguji pilihan *input* gambar yang dipilih oleh *user* apakah sudah sesuai dengan yang diinginkan atau belum.

6.1.1.2 Prosedur Pengujian

Pengujian dilakukan dengan menjalankan antarmuka sistem atau GUI yang sudah dibuat dengan MATLAB. *User* akan memilih pilihan terbaru ataupun pilih manual untuk masukan gambar yang diinginkan. Adapun prosedur dari pengujian dilakukan dengan langkah-langkah berikut ini.

1. Mengambil gambar dari bakteri yang sedang diamati dengan mikroskop digital
2. Menyambungkan kabel USB antara mikroskop dan laptop

3. Menjalankan antarmuka sistem
4. Memilih gambar dengan perbesaran yang berbeda-beda
5. Memilih pilihan *input* gambar yang diinginkan, terbaru atau pilih manual
6. Mengamati output yang dikeluarkan oleh sistem
7. Kesimpulan

6.1.1.3 Hasil dan Analisis

Setelah prosedur pengujian dijalankan, maka didapatkan hasil sebagai berikut:

1. Pada saat *user* melakukan *input* gambar bakteri dengan perbesaran 4x dan 10x maka hasil dapat dilihat pada Gambar 0.2 dan Gambar 0.3

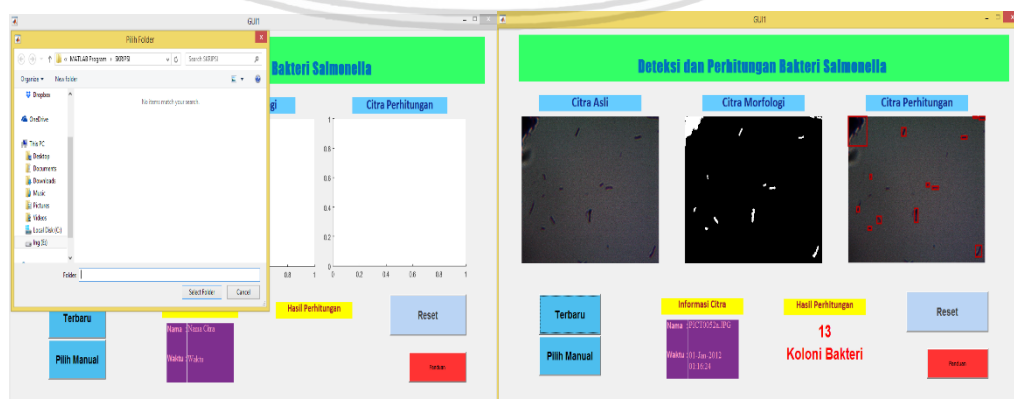


Gambar 0.2 Bakteri Dengan Perbesaran Lensa Objektif 4x



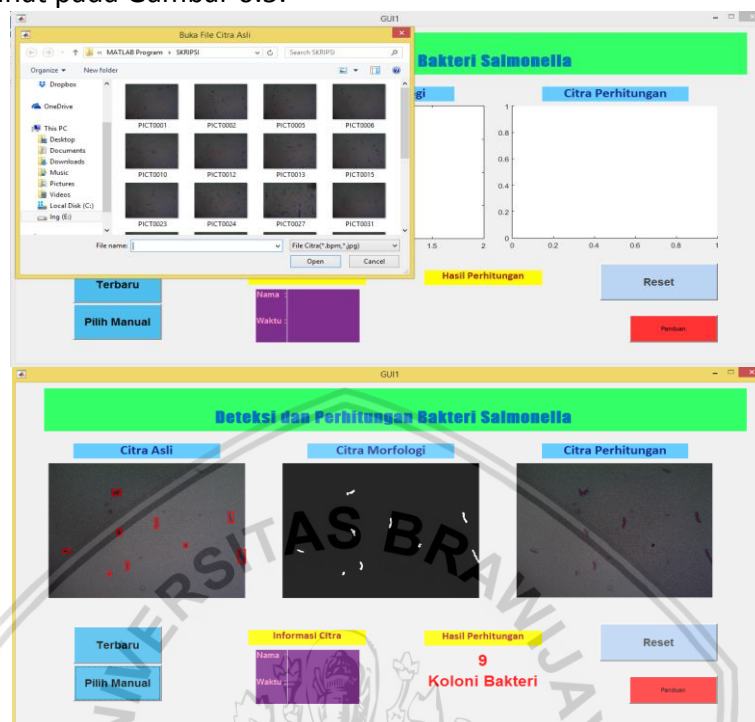
Gambar 0.3 Bakteri Dengan Perbesaran Lensa Objektif 10x

2. Pada saat *user* memilih *input* gambar terbaru, maka sistem akan mengambil gambar terakhir yang diambil oleh mikroskop digital dan akan mengolahnya secara langsung seperti yang terlihat pada Gambar 0.4. Terdapat informasi citra, dari nama dan waktu citra tersebut diakses.



Gambar 0.4 Hasil Pengujian *Input* Gambar Terbaru

3. Pada saat *user* memilih *input* gambar secara manual, maka sistem akan mengeluarkan gambar yang bisa dipilih secara acak oleh *user* seperti yang dapat dilihat pada Gambar 0.5.



Gambar 0.5 Hasil Pengujian *Input* Gambar Secara Manual

Pada hasil pengujian *input* gambar menunjukkan bahwa hasil pengujian *input* gambar baik gambar terbaru ataupun yang dipilih manual oleh *user* telah berhasil dilakukan. Sistem berhasil menjalankan program *input* gambar sesuai dengan yang dipilih oleh *user* dan menampilkan hasil sesuai dengan yang diharapkan. Begitu juga dengan *input* gambar bakteri dengan perbesaran 4x dan 10x. Hasil dari pengujian tersebut kurang baik, pada bakteri dengan perbesaran 4x bentuk koloni bakteri tidak dapat terlihat dengan jelas dan pada bakteri dengan perbesaran 10x banyak koloni bakteri yang tidak terdeteksi karena terlalu kecil, jadi dianggap *noise* oleh sistem.

6.1.2 Pengujian Informasi Gambar

6.1.2.1 Tujuan Pengujian

Tujuan melakukan pengujian pada informasi gambar ialah untuk membuktikan pada saat *user* memilih untuk *input* gambar terbaru benar-benar merupakan gambar terakhir yang diambil oleh mikroskop digital.

6.1.2.2 Prosedur Pengujian

Adapun prosedur dari pengujian dilakukan dengan langkah-langkah berikut ini.

1. Mengambil gambar koloni bakteri dengan menggunakan kamera pada mikroskop digital
2. Menghubungkan mikroskop digital dengan laptop menggunakan kabel USB
3. Menjalankan antarmuka sistem

4. Memilih input gambar terbaru
5. Amati informasi gambar yang diberikan, lalu bandingkan waktu pada gambar yang menjadi input dengan gambar lain yang berada pada satu folder
6. Kesimpulan

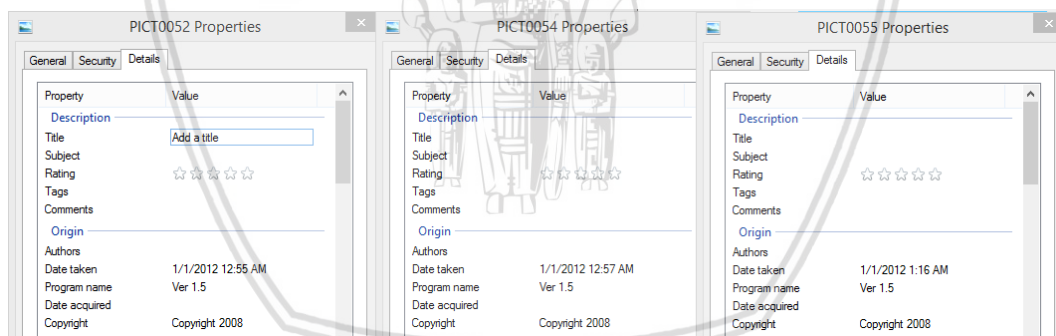
6.1.2.3 Hasil dan Analisis

Setelah semua prosedur pengujian dijalankan maka dapat dilihat pada Gambar 0.6 jika hasil dari informasi gambar yang tertera adalah benar.



Gambar 0.6 Informasi Input Gambar

Hasil tersebut juga sudah dibandingkan dengan gambar yang lain, seperti yang terlihat pada Gambar 0.7. Gambar dengan nama PICT0055 memiliki waktu akses paling akhir daripada PICT0052 dan PICT0054.



Gambar 0.7 Hasil Perbandingan Waktu

6.1.3 Pengujian Reset Program

6.1.3.1 Tujuan Pengujian

Tujuan dilakukan pengujian untuk sub menu *reset* sistem pada antarmuka sistem ialah untuk mengetahui antarmuka sistem dapat menghapus input lama yang sudah berjalan agar *user* dapat menggantinya dengan yang baru.

6.1.3.2 Prosedur Pengujian

Adapun prosedur dari pengujian dilakukan dengan langkah-langkah berikut ini.

1. Menghubungkan mikroskop digital dengan laptop menggunakan kabel USB
2. Menjalankan antarmuka sistem

3. Memilih input gambar yang diinginkan
4. Amati waktu yang dibutuhkan untuk *reset* sistem
5. Kesimpulan

6.1.3.3 Hasil dan Analisis

Berikut ini pada Tabel 6.1 merupakan hasil pengujian dari reset sistem yang dilakukan dari 10x percobaan.

Tabel 0.1 Hasil Pengujian Reset Sistem

No.	Input Gambar	Status
1	Terbaru	Berhasil
2	Terbaru	Berhasil
3	Terbaru	Berhasil
4	Pilih Manual	Berhasil
5	Pilih Manual	Berhasil
6	Terbaru	Berhasil
7	Pilih Manual	Berhasil
8	Pilih Manual	Berhasil
9	Terbaru	Berhasil
10	Pilih Manual	Berhasil

Pada hasil pengujian didapat status berhasil atau dapat melakukan *reset* program dalam setiap jenis *input* yang dipilih *user*.

6.2 Pengujian Akurasi Sistem

6.2.1 Tujuan Pengujian

Tujuan dilakukan pengujian pada akurasi deteksi dan perhitungan koloni bakteri ialah untuk mengetahui ketepatan sistem dalam mendeteksi dan menghitung koloni bakteri *Salmonella* pada input gambar yang ada.

6.2.2 Prosedur Pengujian

Adapun prosedur dari pengujian dilakukan dengan langkah-langkah berikut ini.

1. Menyiapkan isolat bakteri yang akan digunakan untuk pengamatan
2. Membuat preparat bakteri agar memudahkan saat diamati pada mikroskop digital
3. Melakukan teknik pewarnaan gram pada bakteri
4. Melakukan pengamatan pada bakteri dengan mikroskop digital
5. Mengambil gambar bakteri
6. Menyambungkan mikroskop digital dengan laptop menggunakan kabel USB
7. Menjalankan antarmuka sistem
8. Mengamati hasil deteksi dan perhitungan yang ada pada antarmuka sistem
9. Membandingkan hasil deteksi dan perhitungan sistem dengan hasil deteksi dan perhitungan secara manual
10. Kesimpulan

6.2.3 Hasil dan Analisis

Berdasarkan pengujian yang dilakukan untuk melihat akurasi sistem deteksi dan perhitungan koloni bakteri *Salmonella* pada 23 gambar didapatkan beberapa hasil. Ada beberapa gambar dengan tingkat keberhasilan yang tinggi dan ada beberapa gambar dengan tingkat keberhasilan yang rendah. Setelah melakukan perhitungan pada 23 gambar maka didapat hasil seperti pada Tabel 6.2.

Tabel 0.2 Hasil Pengujian Akurasi Sistem Deteksi dan Perhitungan Koloni Bakteri

No.	Nama Gambar	Jumlah Hitung Manual	Jumlah Pengujian	Persentase (%)
1	PICT0001	7	6	85.71
2	PICT0002	9	9	100.00
3	PICT0005	13	11	84.61
4	PICT0006	8	7	87.50
5	PICT0010	20	12	60.00
6	PICT0012	16	7	43.75
7	PICT0013	24	17	70.83
8	PICT0015	12	12	100.00
9	PICT0023	12	11	91.66
10	PICT0024	20	16	80.00
11	PICT0027	38	41	92.10
12	PICT0031	11	9	81.81
13	PICT0033	13	6	46.15
14	PICT0035	8	5	62.50
15	PICT0036	7	9	71.42
16	PICT0039	7	7	100.00
17	PICT0041	25	26	96.00
18	PICT0045	6	5	83.33
19	PICT0047	9	8	88.88
20	PICT0049	10	13	70.00
21	PICT0052	25	29	84.00
22	PICT0053	11	12	90.90
23	PICT0054	40	35	87.50
Rata-rata akurasi sistem				80.81
Rata-rata nilai error sistem				19.18

Hasil dari persentase akurasi sistem deteksi dan perhitungan koloni bakteri dapat dihitung dengan rumus pada persamaan 2.19 berikut ini.

Sebagai contoh, hasil dari pengujian sebuah citra yaitu didapat nilai yang diharapkan atau dalam penelitian ini ialah perhitungan bakteri secara manual sebesar 7 koloni bakteri dan nilai pengujian sebesar 6 koloni bakteri.

$$\text{Persentase keberhasilan} = \frac{38 - |38 - 41|}{38} \times 100\% = 92.10\%$$

Pengujian dilakukan kepada 23 gambar dan didapat persentase rata-rata keberhasilan pada persamaan 2.20 berikut:

$$\text{Rata - rata} = \frac{1858.717}{23} = 80.81\%$$

Hasil persentase keberhasilan sistem deteksi dan perhitungan koloni bakteri ialah sebesar 80.81%. Tingkat keberhasilan deteksi dan perhitungan jumlah koloni bakteri *Salmonella* dipengaruhi oleh beberapa hal seperti pada saat pengambilan gambar dengan kamera mikroskop, teknik pewarnaan yang digunakan, pencahayaan yang digunakan pada mikroskop, serta isolat bakteri yang akan dipakai untuk diamati. Isolat bakteri yang baik ialah bakteri dengan usia 20 – 24 jam, lebih dari 24 jam bentuk bakteri akan terlihat kurang jelas saat diamati dibawah mikroskop.

6.3 Pengujian Waktu Eksekusi Sistem

6.3.1 Tujuan Pengujian

Tujuan pengujian waktu eksekusi sistem ialah untuk mengetahui kecepatan sistem dalam menjalankan semua metode yang digunakan dalam mendeteksi dan melakukan perhitungan koloni bakteri *Salmonella*.

6.3.2 Prosedur Pengujian

Adapun prosedur dari pengujian dilakukan dengan langkah-langkah berikut ini.

1. Menyambungkan mikroskop digital dengan laptop menggunakan kabel USB
2. Menjalankan antarmuka sistem
3. Memilih input gambar yang diinginkan
4. Mengamati waktu yang dibutuhkan sistem sampai menampilkan hasil deteksi dan perhitungan koloni bakteri *Salmonella*.

6.3.3 Hasil dan Analisis

Setelah menjalankan prosedur pengujian maka didapat hasil sesuai dengan Tabel 0.3 berikut ini.

Tabel 0.3 Hasil Pengujian Waktu Eksekusi Sistem

No.	Input	Waktu (detik)
1	Pilih Manual	04.76
2	Terbaru	04.71
3	Pilih Manual	04.43
4	Terbaru	04.59
5	Terbaru	04.56
6	Terbaru	04.57
7	Pilih Manual	04.57
8	Pilih Manual	04.56

9	Terbaru	04.61
10	Pilih Manual	04.60
Rata-rata		04.59

Perhitungan rata-rata dari waktu eksekusi sistem dapat dilihat pada persamaan 2.20

$$Rata - rata = \frac{45.96}{10} = 4.59 \text{ detik}$$

Hasil pengujian dari waktu eksekusi sistem menunjukkan kalau waktu yang diperlukan sistem untuk menjalankan algoritma yang digunakan untuk mendeteksi dan melakukan perhitungan koloni bakteri memiliki rata-rata waktu 4.59 detik.



BAB 7 PENUTUP

Bab ini berisi bahasan mengenai kesimpulan yang diperoleh dari analisis pada setiap pengujian yang dilakukan serta saran untuk pengembangan topik skripsi ataupun pengembangan penelitian lebih lanjut.

7.1 Kesimpulan

Kesimpulan dari rumusan masalah yang didapat berdasarkan pengujian serta hasil analisis yang telah dilakukan terhadap tugas akhir ini ialah sebagai berikut :

1. Untuk dapat melakukan pendeteksian dan perhitungan koloni bakteri *Salmonella* ialah melakukan *image enhancement* dengan metode *high pass filter* sebagai langkah awal untuk memperjelas tepian dari gambar. Setelah itu mengkonversi citra RGB ke dalam citra HSV agar mudah dalam membedakan koloni bakteri *Salmonella* yang sudah diberi pewarnaan gram dengan objek lain. Kemudian mengambil citra saturasi untuk dikonversi kedalam citra biner. Penggunaan citra saturasi pada penelitian ini berdasarkan kebutuhan dan berdasarkan hasil metode experimental, hasilnya akan berbeda dengan penelitian yang lain. Mengubah citra saturasi ke dalam citra biner bertujuan untuk memudahkan saat proses morfologi dilakukan. Setelah citra saturasi terkonversi kedalam citra biner, selanjutnya dilakukan operasi morfologi *closing* untuk memperhalus citra, mengurangi noise, dan menyambungkan objek yang berdekatan. Kemudian dilakukan perhitungan dengan metode object counting untuk mengetahui jumlah koloni bakteri yang ada pada citra tersebut.
2. Pengimplementasian antarmuka sistem pada MATLAB dapat disimpulkan berhasil. Antarmuka sistem dalam bentuk GUI akan menjadi jembatan antara sistem dengan *user*. Desain antarmuka dibuat *user friendly* agar mudah untuk dipahami pada saat menggunakan sistem deteksi dan perhitungan jumlah koloni bakteri.
3. Pada pengujian akurasi sistem, tingkat keberhasilan sistem dalam mendeteksi dan melakukan perhitungan pada koloni bakteri ialah sebesar 80.81%. Akurasi sistem dipengaruhi oleh beberapa hal seperti saat pengambilan gambar, teknik pewarnaan pada bakteri dan juga usia bakteri yang digunakan.
4. Rerata waktu yang dibutuhkan saat melakukan eksekusi sistem ialah 04.59 detik.

7.2 Saran

Saran yang diberikan penulis agar sistem dapat dikembangkan lebih lanjut dengan performa yang lebih baik berdasarkan dari penelitian ialah sebagai berikut:

1. Menggunakan mikroskop digital dengan perbesaran diatas 40x agar hasil gambar koloni bakteri dapat lebih bagus.
2. Sistem dapat dikembangkan dengan menggunakan objek yang lebih bervariasi seperti sel darah merah ataupun objek mikroskopis lainnya.

3. Diharapkan pada penelitian selanjutnya penelitian yang dibuat dapat menentukan tingkat pencemaran suatu objek.



DAFTAR PUSTAKA

- Ari Purno Wahyu Wibowo, R. A. (2016). Perhitungan Jumlah Bakteri Escherichia Coli Dengan Pengolahan Citra Melalui Metode Thresholding dan Counting Morphology. *Jitter*.
- ASUS. (2018, June 14). *X452CP*. Retrieved from <https://www.asus.com/id/Laptops/X452CP/>
- Baby, M., & Philip, L. (2016). Low-Cost Embedded System For Detection Of E.Coli Bacteria In Meat. *IEEE*, 2-3.
- Celestron. (2018, May 15). *LCD DIGITAL MICROSCOPE II*. Retrieved from <https://www.celestron.com/products/lcd-digital-microscope-ii#>
- Efford, N. (2000). *Digital Image Processing a Practical Introduction Using Java*. Essex: Pearson Education Limited.
- Fardiaz, S. (1993). *Analisis Mikrobiologi Pangan*. Jakarta: RajaGrafindo Persada.
- Kadir, A., & Susanto, A. (2013). *Teori dan Aplikasi Pengolahan Citra*. Yogyakarta: CV. ANDI OFFSET (Penerbit Andi).
- Lay, B. W. (1994). *ANALISIS MIKROBA DI LABORATORIUM*. Jakarta: PT RajaGrafindo Persada.
- Mathworks. (2018, May 14). *im2bw*. Retrieved from <https://www.mathworks.com/help/images/ref/im2bw.html>
- Mathworks. (2018, May 14). *Image Types*. Retrieved from https://www.mathworks.com/help/matlab/creating_plots/image-types.html
- Mathworks. (2018, May 15). *imclose*. Retrieved from https://www.mathworks.com/help/images/ref/imclose.html?searchHighlight=IMCLOSE&s_tid=doc_srchtile
- Mathworks. (2018, May 14). *imdilate*. Retrieved from <https://www.mathworks.com/help/images/ref/imdilate.html>
- Mathworks. (2018, May 15). *imerode*. Retrieved from <https://www.mathworks.com/help/images/ref/imerode.html>
- Mathworks. (2018, May 15). *imopen*. Retrieved from <https://www.mathworks.com/help/images/ref/imopen.html>
- Mathworks. (2018, May 14). *rgb2gray*. Retrieved from <https://www.mathworks.com/help/matlab/ref/rgb2gray.html>

- Meera Baby, L. P. (2016). Low-Cost Embedded System For Detection Of E.Coli Bacteria In Meat. *IEEE*, 2-3.
- Pamungkas, A., Adi, K., & Anam, C. (2014). The Automatic Counting Of The Number Of Red Blood Cells And Identification Of Plasmodium Falciparum Phase Using Morphological Operation. 2.
- Peternakan, D. J. (2016, May 8). Retrieved May 8, 2016, from http://www.pertanian.go.id/NAK-2016fix/Prod_DagingAyamRasPedaging_Prop_2016.pdf
- Poeloengan, M., Komala, I., & Noor, S. M. (n.d.). BAHAYA SALMONELLA TERHADAP KESEHATAN. 1-2.
- Putri, M. H., Sukini, & Yodong. (2018, 06 6). *Kementerian Kesehatan Republik Indonesia*. Retrieved from http://bppsdmk.kemkes.go.id/pusdiksdmk/wp-content/uploads/2017/11/mikrobiologi_bab1-9.pdf
- Rangayyan, R. M. (2005). *Biomedical Image Anlysis*. Boca Raton: CRC Press.
- Shih, F. Y. (2009). *Image Processing and Mathematical Morphology*. New York: CRC Press.
- Sianipar, R. H., Mangiri, S. H., & Wiryajati, I. K. (2013). *Matlab Untuk Pemrosesan Citra Digital*. Bandung: INFORMATIKA.
- Sudarno, A. (2016, June 27). Waspada, Daging Ayam di Bogor Mengandung Bakteri Salmonella. *Liputan6*.
- Sugiarto, A. (2006). *Pemrograman GUI dengan MATLAB*. Yogyakarta: CV. ANDI OFFSET.
- Wibowo, A. P., & Andrivani, R. (2016). Perhitungan Jumlah Bakteri Eschericha Coli Dengan Pengolahan Citra Melalui Metode Thresholding dan Counting Morphology. *Jitter*.